Disturbance Compensation Based Controller for an Indoor Blimp Robot

Yue Wang[a,b], Gang Zheng[a,b], Denis Efimov[a,b], Wilfrid Perruquetti[b]

[a] *Inria Lille, 40 Avenue Halley, 59650 Villeneuve d'Ascq, France*
[b] *CRIStAL (CNRS UMR 9189), Ecole Centrale de Lille, 59650 Villeneuve d'Ascq, France*

Abstract

This paper presents the robust controller design for an indoor blimp robot to achieve application such as the surveillance. The commonly used 6 degrees of freedom dynamic model is simplified under reasonable assumptions and decoupled into two independent parts. The blimp simplified horizontal plane movement model is complemented with disturbance terms to ensure the modeling accuracy, then it is transformed to a simpler form for the ease of controller design. Next, the disturbance terms are evaluated by the designed real-time estimator, and the perturbation estimates are compensated in the conceived motion controller for cancellation of the influence of disturbances. The performance and robustness of the disturbance compensation-based controller are verified by both simulations and experiments on the developed blimp robot. Finally, the results prove the feasibility of the blimp robot in indoor surveillance application by stabilizing itself at a fixed position or patrolling along a predefined path.

Keywords: blimp robot, navigation, estimation, uncertainty compensation, robust control.

1. Introduction

Robotics is a quickly developing research field nowadays, and it is an important indicator reflecting human's science and technology development level. Researchers are always interested in the flying robots, which can be classified by their sources of lifting force into two categories: Heavier-Than-Air (HTA) and Lighter-Than-Air (LTA) aircrafts. Among the latter category, it is worth to mention airships for their various advantages compared to other types of aircrafts, such as:

- Ability for VTOL (Vertical Take-Off and Landing), stationary and low speed flight;
- High payload-to-weight ratio;
- Long endurance in air and low energy consumption;
- Low acoustic noise level.

☆Corresponding author: Gang Zheng (gang.zheng@inria.fr)

The blimp robot is a non-rigid airship, which means there are no structures inside the hull, and the shape of the hull is maintained by the internal lifting gas pressure and the strength of the envelope. Therefore, compared to the rigid and semi-rigid airships, the blimp has the advantages of structure simplicity and safe human-robot interaction, it can be made small and suitable for indoor applications such as:

- Long-term surveillance and monitoring;
- Advertising and entertainment;
- Unknown environment exploration and mapping;
- Warehouse goods inventory;
- Pedestrian navigation in large facility;
- Scientific research and education platform.

In this work, a small blimp robot for indoor operation is considered. The modeling and motion control of the robot is studied, and indoor application such as industrial inventory management is validated on the developed blimp robot platform. In order to accomplish the mission, the robot is required to be stabilized at desired position in the workspace, or to follow a predefined path.

To this end, under reasonable assumptions, the motion of blimp is decoupled into independent parts: altitude movement and horizontal plane movement. For altitude stabilization, the earlier obtained solution is used [1]. For the horizontal plane movement, based on real data obtained from the blimp platform in our laboratory, the dynamics is identified as a slider-like nonlinear model complemented with disturbance terms. Next, an output feedback controller with disturbance compensation is designed. Finally, the conceived controllers are experimentally validated.

This paper is organized as follows. In the next section, some related works are introduced and discussed. The blimp robot dynamic model, its simplification and decoupling, and its parameter identification are discussed in section 3. In section 4, one of the decoupled dynamics, the planar movement system, is considered, and an approach to transform the system into a simpler form is presented. Next, in section 5, a real-time estimator of the disturbance term of the system is conceived, then a disturbance compensation based controller is designed, moreover, simulations are made to test its feasibility. Afterwards, we focus on the real blimp, the implementation of designed controller and experiment results are presented in section 6. Finally, conclusion comes in section 7.

2. Related Works

The design and modeling of the giant outdoor airship have been well studied since the last century. Recently, in order to realize autonomous airships, researchers are focusing on the controller design for the airship to achieve tasks such as station keeping, trajectory tracking and path following, etc. Classified by the applied control techniques, we can roughly see two categories when treating the uncertainty (such as external disturbance): the robust control where we design controller to suppress the unknown uncertainty, and the estimation-based controller

where we design a dynamical system to estimate the unknown uncertainty, and compensate it by using the closed-loop controller. For the first category, we can cite the Lyapunov theory based controllers [2, 3, 4, 5, 6], the Line of Sight guidance laws for path following [7, 8], gain scheduling controllers [9, 10], sliding mode controls [11, 12], model predictive controls [13, 14], etc. For the second type, it is also named as active disturbance rejection control in the literature, which has been applied in [15] to control the horizontal trajectory of an airship. In that paper, the authors used input-output linearization method to transform the studied nonlinear model to a controllable and observable part plus the zero dynamics, and then performed a classical PD controller, coupled with the disturbance estimation via the extended state observer. However, the transformation used in [15] is local, which means this transformation is only possible in some well-defined local zones. Such an idea has been also applied to control other types of dynamical systems, such as quad-rotors in [16], where an uncertainty estimator has been designed in frequency domain. This estimator, used to reconstruct the unknown uncertainty, is in fact a low-pass filter, and it therefore can only attenuate the disturbance with the frequency much higher than that of the system, and it does not work for the case when the disturbance's frequency is closed to system's frequency.

To the best of our knowledge, there are fewer works done on the small sized blimps in indoor environments since they have limited payload, and to achieve free movement in indoor cluttered environment, it requires a higher position control accuracy. Here a brief review of the existing works on indoor blimp is presented in chronological order. The work of [17] was an early attempt on the study of indoor blimp robot, they used information provided by sonar system and compared with the desired offsets delivered from the planner to design a controller for blimp, and realized landmark navigation in indoor environment, but the airflow disturbances are not considered in this work. The authors of [18] used camera as sensor of blimp system, and studied the feasibility of transferring the dynamic parameters of the system into the image plane, and realized image-based control of robot to track a quasi-static object. Other researchers also tried blimp control using visual feedback, for instance, the work of [19] designed algorithm to track image regions, and used PID controller for blimp docking and station keeping (keep image window at the center of frame). The authors of [20, 21] used image information and applied extended Kalman filter to do the structure from motion and got position of the blimp, they realized circling control of the blimp robot around a specified target. Later they extended their work to the design of path following controller based on velocity field using inverse optimal control [22, 23], and hovering of blimp against mild wind [24]. The authors of [25] also used on-board camera to realize obstacle avoidance without considering the dynamics of blimp robot. The work [26] studied an insect based neuronal model for collision avoidance and path following, their blimp robot can track a straight line and avoid collision, but the error was relatively large, and the parameters of the neuronal model are trained based on obstacles. The work of [27] used neuronal controllers whose parameters are trained in simulation to map visual input into motor commands, in order to accelerate the movement of flying robot while avoiding collisions. The authors of [28, 29] used model predictive control to handle the constraints of motor saturation and dead-zone, realized point reaching with straight trajectory. The authors of [30] combined their Gaussian Processes (GP) enhanced model to reinforcement learning and designed a controller for blimp yaw and yaw rate control. The work of [31, 32] tackled the control problem of blimp robot with a model-free approach, they applied Monte Carlo learning, and chose GP to approximate the state-action

value function, and by using the ultrasonic (US) sensor information in online learning, their controller achieved blimp altitude control without knowledge of system dynamics or other environment parameters. However, the learning process is rather long. Later, they used the blimp as a platform and studied the problem of localization in indoor environment using US sensors [33], and autonomous navigation in mapped environments based on multi-stage path planning [34]. In the paper of [35], a spherical blimp capable of holonomic motion is designed. The authors of [36] designed vision-based PID controllers for blimp to track human face. In [37], the authors investigated the control for a cubic shape blimp, where the PID controller and the Computed torque controller were presented. However, the influence of external disturbances was not analyzed for those mentioned controllers in [37].

As it can be seen from the literature, some of them did not propose models for indoor blimp robot, which means the researchers use learning-based controllers for the robot motion control, whereas the others design controllers based on complex models, which need accurate parameter identification and raise difficulties for controller design, however the obtained results are not very satisfying.

Since an indoor blimp robot is considered in this work, then it is restricted in size and payload, which limits possible quantity of sensors mounted on the robot and computational complexity of the control and estimation algorithms. That is why in the first part of the work, we intend to study the modeling of the blimp robot, and try to find a novel approach to solve the problem. The idea is to first simplify the complex model under reasonable assumptions, then use the simplified model as a nominal one, and complement it with disturbance terms, next we can design a controller which estimates and compensates the disturbances in real-time [1]. Therefore, the accuracy of control is assured while the complexity is reduced to a minimum. Our developed modeling, estimation and control algorithms are based on real data measurements and experimental validation performed on real platform.

3. Modeling and Parameters Identification

In this section, first the kinematic model of the robot is presented, followed by the commonly used dynamic model. Then some assumptions are proposed to simplify the model and the parameters of the simplified model are identified via tests.

3.1. Kinematic model

The reference frames for the blimp model are shown in Fig. 1.

The frame $\mathcal{F}_n$ is the local navigation frame which is tangent to the Earth surface, its direction is North-East-Down. Since we consider only the operation of blimp robot in indoor environment, the movement of the Earth is ignored, thus the navigation frame $\mathcal{F}_n$ is assumed to be an inertial frame The body-fixed frame $\mathcal{F}_b$ locates its origin at the center of buoyancy (CB) of the blimp, which is also the center of volume (CV) of the hull, the direction of $\mathcal{F}_b$ is forward-right-down.

Due to the fact that the gondola with actuators and other electrical components are mounted on the bottom of the hull, the center of gravity (CG) is located on the $\boldsymbol{Z_b}$ axis of body-fixed frame, therefore denote its coordinate in $\mathcal{F}_b$ as $\boldsymbol{r}_{\boldsymbol{G}}^b = \begin{bmatrix} 0 & 0 & z_G \end{bmatrix}^{\mathrm{T}}$.

The instantaneous linear and angular velocities of the blimp are described in $\mathcal{F}_b$ as

$$\boldsymbol{\xi}^b = \begin{bmatrix} (\boldsymbol{v}^b)^{\mathrm{T}} & (\boldsymbol{\omega}^b)^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} v_x^b & v_y^b & v_z^b & \omega_x^b & \omega_y^b & \omega_z^b \end{bmatrix}^{\mathrm{T}}$$

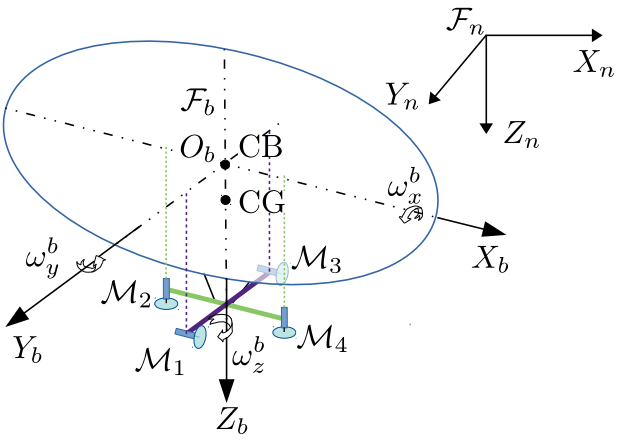

Figure 1: Reference frames for indoor blimp robot

whereas the position and orientation of the blimp with respect to $\mathcal{F}_n$ are expressed as

$$\boldsymbol{\eta}^n = \begin{bmatrix} (\boldsymbol{\eta}_1^n)^{\mathrm{T}} & (\boldsymbol{\eta}_2^n)^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} x^n & y^n & z^n & \phi & \theta & \psi \end{bmatrix}^{\mathrm{T}}$$

where ϕ, θ and ψ are roll, pitch and yaw angle, respectively. The superscripts are used to denote in which reference frame these vectors are defined.

Here the $z - y' - x''$ Tait-Bryan angle is chosen to transform the inertial frame $\mathcal{F}_n$ to the body-fixed frame $\mathcal{F}_b$, to this end, denote the rotation matrix as $\boldsymbol{R}_n^b$. Then from the change of basis equation and the rotational kinematic equation, the blimp kinematic equations can be obtained as:

$$\begin{bmatrix} \dot{\boldsymbol{\eta}}_1^n \\ \dot{\boldsymbol{\eta}}_2^n \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_n^b(\boldsymbol{\eta}_2^n) & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{T}_n^b(\boldsymbol{\eta}_2^n) \end{bmatrix} \begin{bmatrix} \boldsymbol{v}^b \\ \boldsymbol{\omega}^b \end{bmatrix}$$

or

$$\dot{\boldsymbol{\eta}}^n = \boldsymbol{J}(\boldsymbol{\eta}^n)\boldsymbol{\xi}^b \tag{1}$$

where

$$\boldsymbol{R}_n^b(\boldsymbol{\eta}_2^n) = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\phi s_\theta - c_\phi s_\psi & s_\phi s_\psi + c_\phi c_\psi s_\theta \\ c_\theta s_\psi & c_\phi c_\psi + s_\phi s_\psi s_\theta & c_\phi s_\psi s_\theta - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\phi c_\theta \end{bmatrix}$$

and

$$\boldsymbol{T}_n^b(\boldsymbol{\eta}_2^n) = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi/c_\theta & c_\phi/c_\theta \end{bmatrix}$$

The notations $s_{(\cdot)} = \sin(\cdot)$, $c_{(\cdot)} = \cos(\cdot)$, $t_{(\cdot)} = \tan(\cdot)$ are used for compactness.

3.2. Dynamic model

The robot dynamic model linking the accelerations, the forces and moments acting on it can be derived by the Newton-Euler equations of motion [38, 39]. However, due to the fact that the blimp robot is a robot whose density is close to the air, its dynamics is similar to that of underwater vehicles, the added-inertia effects are taken into account [37]. It is shown

that the 6-DOF (degrees of freedom) nonlinear dynamic equations of the blimp motion can be expressed in $\mathcal{F}_b$ as [38]

$$\boldsymbol{M}\dot{\boldsymbol{\xi}}^{\boldsymbol{b}} + \boldsymbol{C}(\boldsymbol{\xi}^b)\boldsymbol{\xi}^b + \boldsymbol{D}(\boldsymbol{\xi}^b)\boldsymbol{\xi}^b + \boldsymbol{g}(\boldsymbol{\eta}^n) = \boldsymbol{\tau}^b \tag{2}$$

where the terms are:

- $\boldsymbol{M}$: the inertia matrix, containing the blimp inertia, and added-inertia terms;
- $\boldsymbol{C}(\boldsymbol{\xi}^b)$: the matrix of the Coriolis and centripetal terms;
- $\boldsymbol{D}(\boldsymbol{\xi}^b)$: the damping matrix;
- $\boldsymbol{g}(\boldsymbol{\eta}^n)$: the vector of restoring forces and moments, including the gravity of the whole robot and the buoyancy generated from the difference between the gas density (air and helium gas in the balloon);
- $\boldsymbol{\tau}^b$: the vector of control inputs, which is used to describe the propulsion forces and moments generated by actuators acting on the blimp in the body-fixed frame.

A brief presentation of these terms is given below.

3.2.1. Inertia matrix

The inertia matrix $\boldsymbol{M}$ contains both the rigid body inertia $\boldsymbol{M}_{\text{RB}}$ and the added-inertia $\boldsymbol{M}_{\text{Added}}$. The rigid body inertia matrix can be written as [38]

$$\boldsymbol{M}_{\text{RB}} = \begin{bmatrix} m\mathfrak{I}_{3\times 3} & -mS(\boldsymbol{r}_{\boldsymbol{G}}^{\boldsymbol{b}}) \\ mS(\boldsymbol{r}_{\boldsymbol{G}}^{\boldsymbol{b}}) & \boldsymbol{I}_{\text{RB}} \end{bmatrix}$$

where m is the mass of the blimp, $\mathfrak{I}_{3\times 3}$ is the identity matrix of dimension 3×3, $\boldsymbol{I}_{\text{RB}}$ is the moment of inertia matrix with respect to CB. Recall that $\boldsymbol{r}_{\boldsymbol{G}}^{\boldsymbol{b}}$ is the coordinate of CG in frame $\mathcal{F}_b$, and $S(\cdot)$ is the skew-symmetric matrix operator.

The added-inertia is caused by the fact that in order to allow the blimp to move in the air, the robot has to push some volumes of the surrounding fluid aside. This phenomenon has a noticeable impact on the blimp, which is a buoyant vehicle and has similar density as air fluid. In result, the phenomenon acts as if the blimp has a bigger inertia than the measured one, which is not accounted in the standard rigid body inertia matrix [38]. Under the assumption that the indoor blimp robot moves slowly and it has three planes of symmetry for the ellipsoid shape hull, the added-inertia can be expressed as a diagonal matrix:

$$\boldsymbol{M}_{\text{Added}} = \text{diag}(\begin{bmatrix} m_{A_x} & m_{A_y} & m_{A_z} & I_{A_x} & I_{A_y} & I_{A_z} \end{bmatrix}^{\text{T}})$$

Then the global inertia can be expressed as:

$$\begin{aligned} \boldsymbol{M} &= \boldsymbol{M}_{\text{RB}} + \boldsymbol{M}_{\text{Added}} \\ &= \begin{bmatrix} m'_x & 0 & 0 & 0 & mz_G & 0 \\ 0 & m'_y & 0 & -mz_G & 0 & 0 \\ 0 & 0 & m'_z & 0 & 0 & 0 \\ 0 & -mz_G & 0 & I'_x & 0 & 0 \\ mz_G & 0 & 0 & 0 & I'_y & 0 \\ 0 & 0 & 0 & 0 & 0 & I'_z \end{bmatrix} \end{aligned} \tag{3}$$

where the added-inertia is included in the diagonal terms.

3.2.2. Coriolis and centripetal forces and moments

From the result of [40], the Coriolis matrix can be derived directly from the inertia matrix

$$\boldsymbol{C}(\boldsymbol{\xi^b}) = \begin{bmatrix} \mathbf{0}_{3\times3} & -S(\boldsymbol{M}_{11}\boldsymbol{v^b} + \boldsymbol{M}_{12}\boldsymbol{\omega^b}) \\ -S(\boldsymbol{M}_{11}\boldsymbol{v^b} + \boldsymbol{M}_{12}\boldsymbol{\omega^b}) & -S(\boldsymbol{M}_{21}\boldsymbol{v^b} + \boldsymbol{M}_{22}\boldsymbol{\omega^b}) \end{bmatrix} \tag{4}$$

where $\boldsymbol{M}_{ij}(i,j=1,2)$ are the four 3×3 sub-matrices of the global inertia matrix $\boldsymbol{M}$, and $S(\cdot)$ as before is the skew-symmetric matrix operator.

3.2.3. Damping forces and moments

Due to air friction, the aerodynamic damping is depended on the velocity of the blimp. In general, there are two types of air frictions: the drag force proportional to the velocity of blimp for laminar flow, and the drag force proportional to the squared velocity for turbulent flow [41, 42]. In the work of [38], the authors modeled the damping forces and moments of slowly moving underwater vehicle by ignoring the terms higher than second order, and proposed a diagonal structure of $\boldsymbol{D}(\boldsymbol{\xi^b})$:

$$\boldsymbol{D}(\boldsymbol{\xi^b}) = -\mathrm{diag}\begin{bmatrix} D_{v_x} + D_{v_x^2}|v_x^b| \\ D_{v_y} + D_{v_y^2}|v_y^b| \\ D_{v_z} + D_{v_z^2}|v_z^b| \\ D_{\omega_x} + D_{\omega_x^2}|\omega_x^b| \\ D_{\omega_y} + D_{\omega_y^2}|\omega_y^b| \\ D_{\omega_z} + D_{\omega_z^2}|\omega_z^b| \end{bmatrix} \tag{5}$$

where D_{v_x}, D_{v_y}, D_{v_z}, D_{ω_x}, D_{ω_y}, D_{ω_z} are the linear damping coefficients, and $D_{v_x^2}$, $D_{v_y^2}$, $D_{v_z^2}$, $D_{\omega_x^2}$, $D_{\omega_y^2}$, $D_{\omega_z^2}$ are the quadratic damping coefficients. According to the authors, the uncoupled damping model (5) works well in case of low speed and highly symmetrical ellipsoid hull [38].

3.2.4. Restoring forces and moments

The lifting force of the blimp is aerostatic, which means it is independent of the flight speed thanks to the helium gas inside the balloon. In practice, the resultant force of buoyancy $\boldsymbol{f_B}$ and gravity $\boldsymbol{f_G}$ will keep the airship upright, thus it is called the restoring force. The gravitational force $\boldsymbol{f_G}$ acts on the CG, whereas the buoyancy force $\boldsymbol{f_B}$ acts at the CB of the blimp robot, which is the origin of $\mathcal{F}_b$, i.e. $\boldsymbol{r_B^b} = \mathbf{0}_{3\times1}$. By using the change of basis equation, in the body fixed frame there is:

$$\boldsymbol{f_G^b} = \boldsymbol{R_b^n}\boldsymbol{f_G^n} = (\boldsymbol{R_n^b})^{\mathrm{T}}\boldsymbol{f_G^n} = (\boldsymbol{R_n^b})^{\mathrm{T}}\begin{bmatrix} 0 \\ 0 \\ f_G \end{bmatrix}$$

Similarly

$$\boldsymbol{f_B^b} = (\boldsymbol{R_n^b})^{\mathrm{T}}\boldsymbol{f_B^n} = (\boldsymbol{R_n^b})^{\mathrm{T}}\begin{bmatrix} 0 \\ 0 \\ -f_B \end{bmatrix}$$

with

$$f_G = mg, \quad f_B = \rho_{\mathrm{air}} V g, \quad V = \tfrac{4}{3}\pi a b^2$$

where m is the mass of the blimp, g is the Earth gravitational acceleration, ρ_{air} is the air density, and V is the volume of ellipsoid shape balloon with semi-axes a and b. Therefore, the restoring forces and moment vector in $\mathcal{F}_b$ is

$$\boldsymbol{g}(\boldsymbol{\eta}^{\boldsymbol{n}}) = -\begin{bmatrix} \boldsymbol{f}_{\boldsymbol{G}}^{\boldsymbol{b}} + \boldsymbol{f}_{\boldsymbol{B}}^{\boldsymbol{b}} \\ \boldsymbol{r}_{\boldsymbol{G}}^{\boldsymbol{b}} \wedge \boldsymbol{f}_{\boldsymbol{G}}^{\boldsymbol{b}} + \boldsymbol{r}_{\boldsymbol{B}}^{\boldsymbol{b}} \wedge \boldsymbol{f}_{\boldsymbol{B}}^{\boldsymbol{b}} \end{bmatrix} \tag{6}$$

where $\wedge$ is the wedge product operator.

3.2.5. Propulsion forces and moments

The propulsive forces of the small indoor blimps are usually generated by motor with propellers, because at low flight speed, the control surface (like rudder and elevator) are not efficient. Moreover, the motors are assumed to be ideal, and the propeller fluxes and motor torques are ignored for the simplicity of modeling. As a consequence, the propulsion forces $\boldsymbol{\tau}^b$ depends only on the motor commands and the installation of motors. For now, we just use the following notation to denote the term $\boldsymbol{\tau}^b$

$$\boldsymbol{\tau}^b = \begin{bmatrix} f_{px} & f_{py} & f_{pz} & \tau_{px} & \tau_{py} & \tau_{pz} \end{bmatrix}^{\mathrm{T}} \tag{7}$$

By substituting the equations (3), (4), (5), (6), and (7) in the equation (2), an explicit form of the dynamic model of the blimp can be obtained (shown in the appendix at the end of this paper).The 6-DOF dynamic model has a very complex form (29), which requires a large amount of accurate experiment data to identify the parameters, but there are still terms that cannot be modeled accurately or that are not considered in the complex form. Therefore, in this work, we intend to simplify the model under reasonable assumptions.

3.3. Simplified model

The following assumptions are proposed in order to simplify the dynamic model.

Assumption 1. *The blimp robot moves slowly (speed less than* 30cm/s*) in the indoor environment.*

Assumption 1 is reasonable because we want to take advantage of the blimp ability for low speed and stationary flight to achieve indoor applications.

Assumption 2. *The blimp roll ϕ and pitch θ angles remain small during movement.*

Under the low speed assumption, the blimp doesn't have any violent moves (unlike quadrotors), moreover Assumption 2 can be verified from the experiment results in section 6. This assumption means that $\phi \simeq \dot{\phi} \simeq \theta \simeq \dot{\theta} \simeq 0$.

Assumption 3. *Under Assumption 2, the propulsion forces of motor $\mathcal{M}_2$ and $\mathcal{M}_4$ only take effect in the $\boldsymbol{Z_n}$-axis direction (the vertical direction of inertial frame). In addition, the propulsion forces of motor $\mathcal{M}_1$ and $\mathcal{M}_3$ only take effect in the plane parallel to $\boldsymbol{O_nX_nY_n}$ (horizontal plane of inertial frame).*

The motors are mounted symmetrically on the gondola as depicted in Fig. 1. The motors $\mathcal{M}_2$ and $\mathcal{M}_4$ are the vertical motors which are in charge of the VTOL of the robot, whereas the motors $\mathcal{M}_1$ and $\mathcal{M}_3$ are responsible for the planar movement control of the blimp.

Assumption 4. *Under Assumption 1, the damping matrix* (5) *is approximated by the linear terms, i.e.*

$$\boldsymbol{D}(\boldsymbol{\xi^b}) = -diag(\begin{bmatrix} D_{v_x} & D_{v_y} & D_{v_z} & D_{\omega_x} & D_{\omega_y} & D_{\omega_z} \end{bmatrix}^T)$$

Assumption 4 is reasonable because the air drag forces caused by turbulent flow can be ignored in low speed movement.

The 6-DOF dynamic model (29) can be simplified under these assumptions, in addition, the roll and pitch dynamics can be ignored. Hence, the simplified model is:

$$\begin{aligned} m'_x\dot{v}^b_x - m'_y\omega^b_z v^b_y - D_{v_x}v^b_x &= f_{px} \\ m'_y\dot{v}^b_y + m'_x\omega^b_z v^b_x - D_{v_y}v^b_y &= f_{py} \\ m'_z\dot{v}^b_z - D_{v_z}v^b_z + (f_B - f_G) &= f_{pz} \\ I'_z\dot{\omega}^b_z + (m'_y - m'_x)v^b_x v^b_y - D_{\omega_z}\omega^b_z &= \tau_{pz} \end{aligned} \tag{8}$$

From the explicit form of the simplified model (8), the following remarks are made.

Remark 1. *The movement in vertical direction (altitude movement) and the planar movement (in horizontal plane) of the blimp studied in this work can be decoupled, which means they are independent of each other.*

Remark 2. *The motion control of the indoor blimp robot can be separated into two sub-problems: altitude movement control and planar movement control, they can be analyzed and solved independently, and then combined together to achieve complete motion control of the blimp.*

Note that the inaccuracy caused by the simplification and decoupling of the blimp motion model will be considered in the complemented disturbance term, this term is estimated in real-time and will be compensated in the controller.

Based on Remark 2, the blimp robot altitude movement controller and horizontal plane movement controller are designed separately, the former one has been already studied in [1], in this paper, we focus on the planar movement controller design and the combination of the two controllers to achieve the complete motion control of the robot.

Consider the first, second and fourth equation in (8), with Assumption 3, on the right hand side we obtain:

$$\begin{cases} f_{px} = f_{\text{left}} + f_{\text{right}} \\ f_{py} = 0 \\ \tau_{pz} = (f_{\text{left}} - f_{\text{right}})l \end{cases}$$

where l is the half distance between the two motors, f_{right} and f_{left} are the propulsive forces generated by motors $\mathcal{M}_1$ and $\mathcal{M}_3$ respectively.

Rearranging the blimp planar movement dynamic model there is:

$$\begin{cases} m'_x\dot{v}^b_x = m'_y v^b_y\omega^b_z + D_{v_x}v^b_x + f_{\text{left}} + f_{\text{right}} \\ m'_y\dot{v}^b_y = -m'_x v^b_x\omega^b_z + D_{v_y}v^b_y \\ I'_z\dot{\omega}^b_z = (m'_x - m'_y)v^b_x v^b_y + D_{\omega_z}\omega^b_z + (f_{\text{left}} - f_{\text{right}})l \end{cases} \tag{9}$$

recall that m'_x, m'_y and I'_z are the apparent mass and moment with respect to different axes; D_{v_x}, D_{v_y} and D_{ω_z} are the corresponding damping coefficients.

In fact, after simplification and decoupling, the blimp movement in the horizontal plane (9) is similar to a slider which moves on a 2D horizontal plane and whose velocities are not restricted (unlike nonholonomic wheeled vehicles) [43, 44, 45].

Remark 3. *In the third equation of* (9)*, it can be seen that the difference between apparent mass in the lateral and longitudinal direction will cause an unstable yaw moment during the moving forward motion. To simplify the dynamic model, recall that the blimp is supposed to move slowly, and the global mass terms* m'_x *and* m'_y *are assumed to be approximately equal, i.e.* $m'_x \simeq m'_y = m_{Horiz}$*, thus this unstable yaw moment term disappears. Once more, the inaccuracy caused by the hypothesis will be considered in the complemented disturbance term and compensated in the designed controller.*

From the kinematic model (1) and under the assumption proposed in Remark 3, we can obtain:

$$\begin{cases} \ddot{x} = c_\psi \frac{f_{\text{left}}+f_{\text{right}}}{m_{\text{Horiz}}} + (a_x c_\psi^2 + a_y s_\psi^2)\dot{x} + (a_x c_\psi s_\psi - a_y c_\psi s_\psi)\dot{y} \\ \ddot{y} = s_\psi \frac{f_{\text{left}}+f_{\text{right}}}{m_{\text{Horiz}}} + (a_y c_\psi^2 + a_x s_\psi^2)\dot{y} + (a_x c_\psi s_\psi - a_y c_\psi s_\psi)\dot{x} \\ \ddot{\psi} = \frac{(f_{\text{left}}-f_{\text{right}})l}{I'_z} + a_\psi \dot{\psi} \end{cases}$$

where $a_x = D_{v_x}/m_{\text{Horiz}}$, $a_y = D_{v_y}/m_{\text{Horiz}}$, $a_\psi = D_{\omega_z}/I'_z$. Note that the superscript $(\cdot)^n$ for x and y is omitted for simplicity of notation.

For clarity of the expression, rewrite the blimp simplified planar movement model as:

$$\begin{cases} \ddot{x} = c_\psi b u + \kappa_1(a_x, a_y, \psi)\dot{x} + \kappa_3(a_x, a_y, \psi)\dot{y} \\ \ddot{y} = s_\psi b u + \kappa_2(a_x, a_y, \psi)\dot{y} + \kappa_3(a_x, a_y, \psi)\dot{x} \\ \ddot{\psi} = b_\psi v + a_\psi \dot{\psi} \end{cases} \tag{10}$$

where u and v are the two control inputs, with $u = u_{\text{left}} + u_{\text{right}}$ and $v = u_{\text{left}} - u_{\text{right}}$; u_{right} and u_{left} are respectively the value of command signal for right and left motors; b and b_ψ are the coefficients related to the control inputs; coefficients $\kappa_1(a_x, a_y, \psi) = a_x c_\psi^2 + a_y s_\psi^2$, $\kappa_2(a_x, a_y, \psi) = a_y c_\psi^2 + a_x s_\psi^2$ and $\kappa_3(a_x, a_y, \psi) = a_x c_\psi s_\psi - a_y c_\psi s_\psi$.

In order to obtain a simple nominal model, by taking into account the special forms of κ_1, κ_2 and κ_3, it is assumed that the two damping coefficients in lateral and longitudinal direction of the nominal model are equal, i.e. $a_x = a_y$. Depending on the shape of blimp, this assumption might not be always valid. But we will show hereafter that, for the case where $a_x \neq a_y$, we can always set $a_x = \bar{a}_y$ for the nominal model, and then treat $a_y - \bar{a}_y$ as the modeling uncertainty which will be estimated via observer and will be compensated by the designed closed-loop controller. In this sense, it is not critical for our method to have a nominal model with $a_x = a_y$. Thus, in equation (10), we set the terms $\kappa_1(a_x, a_y, \psi) = \kappa_2(a_x, a_y, \psi) = a_x = a_y$, and $\kappa_3(a_x, a_y, \psi) = 0$. Based on the obtained nominal model, the following will present the way to identify those unknown parameters.

3.4. Parameter identification

A real blimp robot is developed by us as shown in Fig. 9 and 10, the experiment platform is presented in section 6. Parameter identification is performed on this robot for its horizontal plane movement model (10). Due to the payload limit of the balloon, only low weight sensors can be mounted on the robot control board, which increases the difficulties for the blimp autonomous localization in the indoor environment. Therefore, the camera motion capturing system OptiTrack is selected to localize the blimp in the testing room to track the robot and obtain its position and orientation measurements. The OptiTrack system uses infrared waves to capture the reflective markers mounted on blimp control board, and solves the pose of the

robot at a rate of 100 frames per second, and the precision for position measurement is 1mm [46].

The identification process is performed by two different types of tests, where the first one is for identifying a_x, a_y, b by the forward and backward movement of the blimp without turning, and the second one is for identifying a_ψ and b_ψ by the spinning motion without changing position.

In the first type of tests, the blimp is first manually pushed forward and moves along a straight line in its $\boldsymbol{X_b}$-axis direction, and no command is sent to the horizontal motors $\mathcal{M}_1$ and $\mathcal{M}_3$, i.e. $u = 0$, $v = 0$, the robot decelerates by the damping forces, its position is measured by OptiTrack system. The equation is

$$\begin{cases} \ddot{x} = a_x \dot{x} \\ \ddot{y} = a_y \dot{y} \end{cases}$$

Then the homogeneous differentiator (HOMD) is used to get the first- and second-order time-derivatives of the position $\dot{x}$, $\ddot{x}$, $\dot{y}$ and $\ddot{y}$. The detailed presentation of the used differentiator can be found in [1]. By applying least squares method, the parameter $a_x = a_y$ can be identified. The result is

$$a_x = a_y = -0.24$$

In the next step, the identified parameters for air friction is used in the nominal model, and constant command is given at same time to the motors $\mathcal{M}_1$ and $\mathcal{M}_3$, i.e. $u > 0$ and $v = 0$, the propulsion forces make the blimp move forward, and measurements of position and orientation in the straight line part (where ψ can be assumed invariant) of the trajectories are used to estimate b. The equation is

$$\begin{cases} \ddot{x} = c_\psi b(\text{sign}(u))u + a_x \dot{x} \\ \ddot{y} = s_\psi b(\text{sign}(u))u + a_y \dot{y} \end{cases}$$

Here the coefficient for motor input b is denoted by $b(\text{sign}(u))$ due to the fact that the blimp horizontal motor $\mathcal{M}_1$ and $\mathcal{M}_3$ have different efficiencies when they rotate in clockwise/counter-clockwise direction given same input value but opposite signs.

Again the HOMD differentiator is used to get the first and second-order time-derivatives of the position $\dot{x}$, $\ddot{x}$, $\dot{y}$ and $\ddot{y}$. The ψ is measured by OptiTrack and in the straight line part, ψ is approximately constant, then by least squares method, the parameter b can be identified:

$$b(u) = \begin{cases} 0.0822 & \text{if} \quad u \geq 0 \\ 0.0527 & \text{if} \quad u < 0 \end{cases}$$

In the second type of tests, the coefficients a_ψ and b_ψ are identified. Similarly, the blimp is first given a manual torque to spin around its $\boldsymbol{Z_b}$-axis without change of position, the horizontal motors are given zero commands. The yaw angle is measured by the camera system, and the blimp decelerates by the damping torque $a_\psi \dot{\psi}$. Then $\ddot{\psi}$ and $\dot{\psi}$ are calculated by differentiator and then parameter a_ψ is identified as

$$a_\psi = -0.20$$

Then after a_ψ is identified and substituted in the yaw dynamic equation in nominal model, the motors $\mathcal{M}_1$ and $\mathcal{M}_3$ are given step input with same value but opposite signs at same

moment, i.e. $u = 0$ and $v > 0$ (or $v < 0$), to make the blimp rotating while not changing its position in horizontal plane. From this test the parameter b_ψ is identified as

$$b_\psi = 0.0668$$

In summary, the planar movement model parameter identification result for our blimp robot is:

$$\begin{aligned} & a_x = a_y = -0.24 \\ & b(u) = \begin{cases} 0.0822 & \text{if} \quad u \geq 0 \\ 0.0527 & \text{if} \quad u < 0 \end{cases} \\ & a_\psi = -0.2 \\ & b_\psi = 0.0668 \end{aligned} \tag{11}$$

It is worth to mention that in the parameter identification process, none of the physical quantities are measured directly, such as the rigid body mass, the propulsion force, etc. Instead, a relation between the time-derivatives of position (and yaw angle) and the input is established and the parameters are identified via real test measurements. Therefore, the nominal model can reflect sufficiently well the actual performance of the blimp when it is in motion.

4. Planar Movement System Description

For blimp decoupled movement in horizontal plane, the simplified model (10) is used as a nominal model. Then in order to ensure the accuracy of control, disturbance terms are added to the nominal model, which represent the errors between nominal model and real one, they include the errors caused by:

- Nominal model parameter identification inaccuracy;
- Airflow perturbation to the balloon;
- Ignored motor dynamics during modeling;
- Coupling of the two "decoupled" motions;
- Other environmental disturbances, which are impossible to be accurately modeled.

Thus, after identification of the parameters, the planar movement nominal model (10) complemented with disturbance terms becomes:

$$\begin{cases} \ddot{x} = c_\psi b u + a_x \dot{x} + d_x \\ \ddot{y} = s_\psi b u + a_y \dot{y} + d_y \\ \ddot{\psi} = b_\psi v + a_\psi \dot{\psi} + d_\psi \end{cases} \tag{12}$$

where $a_x = a_y$, moreover, d_x, d_y and d_ψ are the disturbance terms which are estimated on-line, they are assumed to be small, bounded and smooth.

Note that the system (12) is under-actuated, since it has 3 configurations x, y and ψ but only two control inputs u, v. A quick verification shows that this system doesn't satisfy the conditions of the Brockett's Theorem [47]. Therefore, there exists no smooth time-independent static state feedback controller which makes the origin of (12) asymptotically stable.

The scheme adopted in this paper is to design the following switched controller

$$U = \begin{cases} U_{xy}, & \text{if } \sqrt{(x-x_r)^2+(y-y_r)^2} \geq \varepsilon_{xy} \\ U_{\psi}, & \text{otherwise} \end{cases}$$

where U_{xy} is the controller dedicated to regulate (x, y) if the relative distance between the current position of blimp and its desired position (x_r, y_r) is bigger than a prescribed threshold ε_{xy}, and U_{ψ} represents the controller to drive ψ to its desired orientation when the blimp is sufficiently close to its desired position.

From the third equation of (12), it is easy to see that the orientation ψ is independent of u, and it can be written into the following linear form with disturbance:

$$\dot{\Psi} = \begin{bmatrix} 0 & 1 \\ 0 & a_{\psi} \end{bmatrix} \Psi + \begin{bmatrix} 0 \\ b_{\psi} \end{bmatrix} (v + \tfrac{d_{\psi}}{b_{\psi}}) \tag{13}$$

where $\Psi = \left[\psi, \dot{\psi}\right]^T$. This dynamics is in the form which will be discussed in Section 4.2 (Eq. (21)). There, an observer will be constructed to estimate the disturbance d_{ψ}, noted as $\hat{d}_{\psi}$, and then a robust controller will be designed to compensate the disturbance.

In order to avoid the redundancy, the following will firstly focus on the design of U_{xy}. After that we will show that the same methodology can be used to design U_{ψ}.

4.1. Transformation

This subsection presents an approach to transform the under-actuated system, via a coordinate transformation [48],to a simpler one for controller design. Considering the dynamics of a point Q on the $\boldsymbol{X}_b$-axis of the blimp body-fixed frame (see Fig. 2), and apply control to regulate its position. Specifically, the distance between Q and O_b is denoted by q, define the coordinates of the point Q in the horizontal plane of navigation frame $\mathcal{F}_n$ as (s, r), knowing that the coordinates of O_b in the horizontal plane is (x, y), we get:

$$\begin{bmatrix} s \\ r \end{bmatrix} = \begin{bmatrix} x + q\cos\psi \\ y + q\sin\psi \end{bmatrix} \tag{14}$$

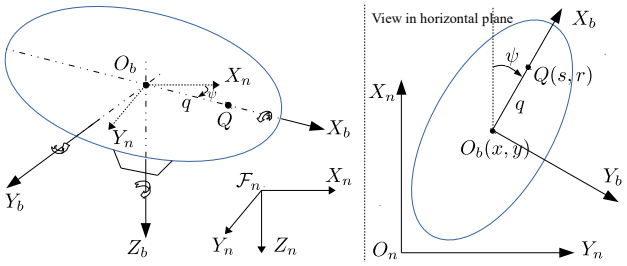

Figure 2: Blimp coordinates in horizontal plane

Taking the second-order time-derivative of s and r, then substituting $\ddot{x}$, $\ddot{y}$ and $\ddot{\psi}$ from (12), there is

$$\begin{cases} \ddot{s} = a_x\dot{x} + c_{\psi}bu - qc_{\psi}\dot{\psi}^2 - qs_{\psi}(a_{\psi}\dot{\psi} + b_{\psi}v) \\ \qquad\qquad + d_x - qs_{\psi}d_{\psi} \\ \ddot{r} = a_y\dot{y} + s_{\psi}bu - qs_{\psi}\dot{\psi}^2 + qc_{\psi}(a_{\psi}\dot{\psi} + b_{\psi}v) \\ \qquad\qquad + d_y + qc_{\psi}d_{\psi} \end{cases} \tag{15}$$

where $a_x = a_y$. Substituting $\dot{x}$ and $\dot{y}$ in (15) by:

$$\begin{aligned}\dot{x} &= \dot{s} + qs_\psi\dot{\psi} \\ \dot{y} &= \dot{r} - qc_\psi\dot{\psi}\end{aligned}$$

we get:

$$\begin{cases} \ddot{s} = a_x\dot{s} + (c_\psi bu - qs_\psi b_\psi v) + (d_x - qs_\psi d_\psi) \\ \qquad\qquad + \left(a_x qs_\psi\dot{\psi} - qc_\psi\dot{\psi}^2 - qs_\psi a_\psi\dot{\psi}\right) \\ \ddot{r} = a_y\dot{r} + (s_\psi bu + qc_\psi b_\psi v) + (d_y + qc_\psi d_\psi) \\ \qquad\qquad + \left(-a_y qc_\psi\dot{\psi} - qs_\psi\dot{\psi}^2 + qc_\psi a_\psi\dot{\psi}\right) \end{cases} \tag{16}$$

Let the new control input be:

$$U_{xy} = \begin{bmatrix} \tilde{u} \\ \tilde{v} \end{bmatrix} = \Theta \begin{bmatrix} u' \\ v \end{bmatrix} = \Theta \begin{bmatrix} b(u)u \\ v \end{bmatrix} \tag{17}$$

with

$$\Theta = \begin{bmatrix} c_\psi & -qs_\psi b_\psi \\ s_\psi & qc_\psi b_\psi \end{bmatrix}$$

Note that the coefficient b with respect to control input u is written as $b(u)$ according to the parameter identification result (11).

Remark 4. *The matrix Θ is invertible for $q \neq 0$. In practice, the q is chosen as a small constant but not too small, such that the matrix Θ is neither singular nor ill-conditioned, in our tests the distance is set as $q = 5cm$. In fact, since the size of the blimp balloon (length $110cm$) is much bigger than the chosen q, thus the position error (between point O_b and point Q) is acceptable for our applications.*

Remark 5. *With the proposed coordinate transformation method, the problem of controlling the exact position (x, y) and orientation ψ of the robot is transformed to a practical control problem of the position of point $Q(s, r)$ which is close to the robot body-fixed frame center O_b. Therefore, when the position of point Q is regulated to the desired location, the robot center (x, y) lies on a circle which is centered at (s, r) and with a radius q.*

For $q \neq 0$, the matrix Θ is invertible thus $\begin{bmatrix} u' & v \end{bmatrix}^{\mathrm{T}}$ can be solved from U, then depending on the sign of u', u can be obtained by:

$$u = u'/b(u) \tag{18}$$

$b(u)$ is chosen from (11), note that $b(u)$ is always positive.

4.2. System description

In (16), denote

$$\Delta = \begin{bmatrix} \Delta_1 \\ \Delta_2 \end{bmatrix} = \begin{bmatrix} a_x qs_\psi\dot{\psi} - qc_\psi\dot{\psi}^2 - qs_\psi a_\psi\dot{\psi} \\ -a_y qc_\psi\dot{\psi} - qs_\psi\dot{\psi}^2 + qc_\psi a_\psi\dot{\psi} \end{bmatrix} \tag{19}$$

and disturbance term

$$D = \begin{bmatrix} d_s \\ d_r \end{bmatrix} = \begin{bmatrix} d_x - qs_\psi d_\psi \\ d_y + qc_\psi d_\psi \end{bmatrix} \tag{20}$$

From (16), the state vector is chosen as $X = \begin{bmatrix} s & \dot{s} & r & \dot{r} \end{bmatrix}^{\mathrm{T}}$, hence the system for blimp planar movement can be written in state space form as:

$$\begin{cases} \dot{X} = AX + B(U_{xy} + \Delta + D) \\ y = CX \end{cases} \tag{21}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & a_x & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & a_y \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Note that $a_x = a_y$ and the matrix A is constant. Therefore, the blimp robot horizontal plane movement control system (21) studied in this work is considered as a *nonlinear system complemented with uncertain bounded disturbances*.

5. Planar Movement Controller

As it is presented in the previous section, with the coordinate transformation approach, the under-actuated system (12) is transformed to a simpler form for the ease of controller design, and from Remark 5, it is noticed that the exact control problem of blimp position in the horizontal plane and yaw angle is transformed to a practical control problem of only the position of point $Q(s, r)$. In order to design a disturbance compensation based controller for the system (21), firstly, the disturbance term D needs to be estimated.

5.1. Disturbance estimation

The disturbance term D in system (21) represents the error between nominal model (10) and blimp real situation. In order to estimate it, a filter is designed:

$$\begin{cases} \dot{X}_{\text{fil}} = AX_{\text{fil}} + B(U_{xy} + \Delta) + L(y - y_{\text{fil}}) \\ y_{\text{fil}} = CX_{\text{fil}} \end{cases} \tag{22}$$

where L is the gain of filter. Let $e_{\text{fil}} = X - X_{\text{fil}}$, its dynamics is:

$$\dot{e}_{\text{fil}} = (A - LC)e_{\text{fil}} + BD \tag{23}$$

Assume

$$L = \begin{bmatrix} l_{11} & l_{12} \\ l_{21} & l_{22} \\ l_{31} & l_{32} \\ l_{41} & l_{42} \end{bmatrix}$$

Then, in order to assure the stability, the filter matrix $A - LC$ needs to be Hurwitz, and the gain L can be chosen to make the error converges at least 5-10 times faster than the closed-loop dynamics (21) [49].

From (23), there is:

$$\begin{cases} \dot{e}_1 = -l_{11}e_1 + e_2 - l_{12}e_3 \\ \dot{e}_2 = -l_{21}e_1 + \kappa_1 e_2 - l_{22}e_3 + d_s \\ \dot{e}_3 = -l_{31}e_1 - l_{32}e_3 + e_4 \\ \dot{e}_4 = -l_{41}e_1 - l_{42}e_3 + \kappa_2 e_4 + d_r \end{cases} \tag{24}$$

From the first and third equation, we have:

$$\begin{cases} e_2 = \dot{e}_1 + l_{11}e_1 + l_{12}e_3 \\ e_4 = \dot{e}_3 + l_{31}e_1 + l_{32}e_3 \end{cases}$$

Taking time-derivative on both sides:

$$\begin{cases} \dot{e}_2 = \ddot{e}_1 + l_{11}\dot{e}_1 + l_{12}\dot{e}_3 \\ \dot{e}_4 = \ddot{e}_3 + l_{31}\dot{e}_1 + l_{32}\dot{e}_3 \end{cases}$$

Substituting e_2, e_4, $\dot{e}_2$ and $\dot{e}_4$ in the second and fourth equation of (24), the expression of disturbance estimation is obtained:

$$\begin{cases} \hat{d}_s = \ddot{e}_1 + (l_{11} - \kappa_1)\dot{e}_1 + (l_{21} - \kappa_1 l_{11})e_1 \\ \qquad\quad + l_{12}\dot{e}_3 + (l_{22} - \kappa_1 l_{12})e_3 \\ \hat{d}_r = \ddot{e}_3 + (l_{32} - \kappa_2)\dot{e}_3 + (l_{42} - \kappa_2 l_{32})e_3 \\ \qquad\quad + l_{31}\dot{e}_1 + (l_{41} - \kappa_2 l_{31})e_1 \end{cases} \tag{25}$$

As blimp body-fixed frame origin O_b position (x, y) can be measured by the OptiTrack system, with the relation (14), position of control point Q can be evaluated directly, thus $e_1 = s - s_{\text{fil}}$, $e_3 = r - r_{\text{fil}}$ are known. Therefore only $\dot{e}_1$, $\ddot{e}_1$, $\dot{e}_3$ and $\ddot{e}_3$ need to be evaluated to get the estimation of disturbance $\hat{d}_s$ and $\hat{d}_r$ [1, 50]. For this purpose, the HOMD differentiator is applied again. Here a third-order HOMD differentiator is used in order to get higher estimation accuracy [51, 52].

5.2. Disturbance compensation based controller

For the robust control of point $Q(s, r)$ in horizontal plane, let $\begin{bmatrix} s_{\text{ref}} \\ r_{\text{ref}} \end{bmatrix}$ be the reference point (or trajectory) with the error term:

$$e = \begin{bmatrix} e_s \\ e_r \end{bmatrix} = \begin{bmatrix} s - s_{\text{ref}} \\ r - r_{\text{ref}} \end{bmatrix} \tag{26}$$

The following theorem is obtained:

Theorem 1. *For system* (21)*, if the disturbance compensation based controller is chosen as*

$$U_{xy} = \begin{bmatrix} \tilde{u} \\ \tilde{v} \end{bmatrix} = \begin{bmatrix} -\Delta_{c1} - \hat{d}_s - K_p e_s - K_d \dot{e}_s + \ddot{s}_{\text{ref}} \\ -\Delta_{c2} - \hat{d}_r - K_p e_r - K_d \dot{e}_r + \ddot{r}_{\text{ref}} \end{bmatrix} \tag{27}$$

where $\hat{d}_s$*,* $\hat{d}_r$ *are estimated with equation (25),* $\dot{e}_s = \dot{\hat{s}} - \dot{s}_{\text{ref}}$*,* $\dot{e}_r = \dot{\hat{r}} - \dot{r}_{\text{ref}}$*, the estimates* $\hat{s}$ *and* $\hat{r}$ *are also obtained by HOMD differentiator,* K_p*,* K_d *are the gains of controller, and*

$$\Delta_c = \begin{bmatrix} \Delta_{c1} \\ \Delta_{c2} \end{bmatrix} = \begin{bmatrix} \kappa_1 \dot{x} + (-q c_\psi \dot{\psi} - q s_\psi a_\psi)\dot{\psi} \\ \kappa_2 \dot{y} + (-q s_\psi \dot{\psi} + q c_\psi a_\psi)\dot{\psi} \end{bmatrix}$$

then the position of control point Q *in horizontal plane converges to the desired one with exponential convergence rate.*

Proof. Take the second order time-derivative of equation (26), and with (15) we have:

$$\ddot{e} = \begin{bmatrix} \ddot{e}_s \\ \ddot{e}_r \end{bmatrix} = \begin{bmatrix} \ddot{s} - \ddot{s}_{\text{ref}} \\ \ddot{r} - \ddot{r}_{\text{ref}} \end{bmatrix} = \Delta_c + U_{xy} + D - \begin{bmatrix} \ddot{s}_{\text{ref}} \\ \ddot{r}_{\text{ref}} \end{bmatrix}$$

Using the controller proposed in Theorem 1, the error dynamics becomes:

$$\begin{cases} \ddot{e}_s + K_d \dot{e}_s + K_p e_s = d_s - \hat{d}_s \\ \ddot{e}_r + K_d \dot{e}_r + K_p e_r = d_r - \hat{d}_r \end{cases}$$

As presented in [1, 52], with HOMD differentiator, the estimation of d_s and d_r converges in finite-time. Therefore, by adjusting the controller gains K_p and K_d, we can set the poles of error dynamic equation anywhere we want, which guarantee the exponential convergence of e_s and e_r to zero due to BIBS (Bounded Input Bounded State) property of these two second-order linear systems. □

Therefore, with the disturbance compensation based controller designed in Theorem 1, the blimp planar movement closed-loop system is linearized and disturbance terms are compensated in finite-time, thus the control point Q on the blimp is supposed to be successfully stabilized at desired position or following a predefined trajectory in the horizontal plane.

For the blimp to track a trajectory, the problem remains to plan a path from the initial position to the final position. One available method is to use time-polynomial as trajectory, and decide the parameters by boundary conditions [53].

Remark 6. *For a system of the form (21) with unknown disturbances, we proposed in the above to design, first, an observer of the type (22), combining with the HOMD differentiator, to estimate the disturbance, and then design a robust controller* U_{xy} *as in (27) to compensate the influence of disturbance. Note that the dynamics of the orientation, described by (13) is exactly of the form (21), therefore it is easy to design a structurally similar controller* U_ψ*, by simply following the same procedure as for* U_{xy}*, to realize the robust control of the orientation of the blimp.*

5.3. Simulation

Simulations are made via MATLAB Simulink to validate the designed disturbance compensation based controller for the blimp planar movement control. The parameters for simulation are set as follows:

- the time step is set as 0.01s, to simulate the measurement frequency of the experimental platform which is 100Hz.
- the parameter q is set as 5, which means the control point Q locates 5cm away from the blimp body-fixed frame center O_b.
- for the filter (22), the parameters are set as the nominal model identification result (11), which indicates $\kappa_1 = \kappa_2 = a_x = a_y$ and $\kappa_3 = 0$.
- for the disturbance estimator (25), the HOMD differentiator used to estimate $\dot{e}_1$, $\ddot{e}_1$, $\dot{e}_3$ and $\ddot{e}_3$ has the gains:

  $$k_1 = 50, \quad k_2 = 400, \quad k_3 = 200, \quad k_4 = 10$$

  And here a third-order HOMD differentiator is used for first- and second-order time-derivative estimation for better accuracy [51, 52].
- the HOMD differentiator for $\dot{s}$ and $\dot{r}$ estimation (used in controller) has the gains:

  $$k_1 = 50, \quad k_2 = 400, \quad k_3 = 200$$

  Here a second-order HOMD differentiator is used for first-order time-derivative estimation for similar reason.
- the controller gain is chosen as

  $$K_p = 0.1, \quad K_d = 0.7$$

  which makes the closed-loop system having two poles of $p_1 = -0.2$, $p_2 = -0.5$. The poles are not too big because of the assumption that the blimp moves slowly.
- the gain L for the filter is set as:

  $$L = \begin{bmatrix} 3.76 & 0 \\ 3.0976 & 0 \\ 0 & 3.76 \\ 0 & 3.0976 \end{bmatrix}$$

  which makes the poles of filter to be $\begin{bmatrix} -2 & -2 & -2 & -2 \end{bmatrix}$, which is faster than closed-loop system.

Next, two types of tasks for the blimp horizontal plane movement control are simulated: point stabilization and tracking of predefined trajectories, to verify the performance of the designed disturbance compensation based controller.

5.3.1. Point stabilization

In this simulation, the blimp is supposed to reach set point

$$s_{\text{ref}} = 200, \quad r_{\text{ref}} = 200$$

and stabilized there. The initial state of blimp is set to be

$$s_0 = r_0 = 0, \quad \psi_0 = -\pi/4$$

which means the control point Q is initialized at origin, and the initial direction of blimp is perpendicular to the line connecting initial and the final points. The disturbance terms are set as:

$$\begin{aligned} d_x(t) &= \begin{cases} 2\sin(0.2t) + 1, & t < 150 \\ -4\sin(0.1t) - 2, & t \geq 150 \end{cases} \\ d_y(t) &= \begin{cases} -4\sin(0.1t) - 2, & t < 100 \\ 2\sin(0.2t) + 1, & t \geq 100 \end{cases} \\ d_\psi(t) &= 0 \end{aligned}$$

Moreover a white-noise generated by "Gaussian Noise Generator" block of Simulink with variance set as 0.001 is added to both d_x and d_y. According to relation (20), as $d_\psi = 0$, there is $d_s = d_x$ and $d_r = d_y$. In addition, the "real system" in simulation has parameters:

$$\begin{aligned} & a_x = a_y = -0.26 \\ & a_\psi = -0.16 \\ & b(u) = \begin{cases} 0.09 & \text{if} \quad u \geq 0 \\ 0.045 & \text{if} \quad u < 0 \end{cases} \\ & b_\psi = 0.06 \end{aligned}$$

which are different to the nominal model parameters (11) to simulate the parameter identification inaccuracy.

The results of point Q position errors e_s and e_r are shown in Fig. 3.

It is shown that under time-varying disturbances d_x, d_y, and parameter identification inaccuracy, the designed controller robustly stabilizes robot at desired position, and the error of position oscillates only in a small region near the origin, which is acceptable in our application scenario. In addition, when the external perturbation has sudden changes (at moment 100s and 150s), the position of the blimp in horizontal plane does not move far from the desired one, which shows the robustness of the designed controller.

As for the disturbance estimation results, they are shown in Fig. 4.

It is worth to mention that the estimated disturbances $\hat{d}_s$ and $\hat{d}_r$ reflect the difference between the nominal model and the real one, in this test, it includes not only the added perturbation d_s and d_r, but also the difference due to parameter identification inaccuracy, that is why in the Fig. 4, the curves are not completely superposed. However, it can be observed that when the parameters of "real system" do not differ largely from the nominal model, the estimated $\hat{d}_s$ and $\hat{d}_r$ follows the variation trend of the added d_s and d_r, even with the sudden change at moment 100s and 150s.

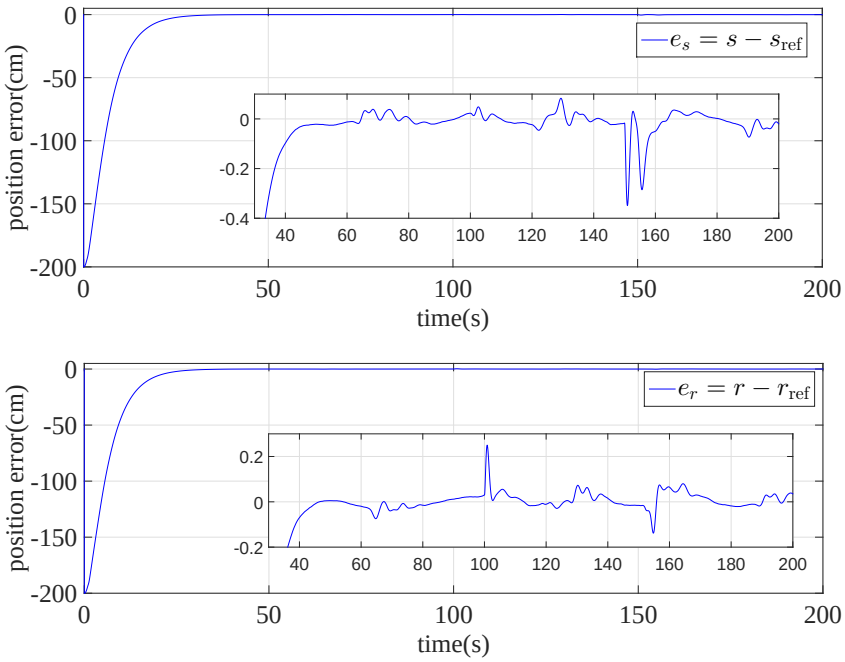

Figure 3: Point stabilization simulation Test-Q position error result

Furthermore, to compare the effectiveness of the proposed disturbance compensation based controller, simulations are made with the classic PID controller. The PID controller is designed as (see (27)):

$$U_{xy} = \begin{bmatrix} \tilde{u} \\ \tilde{v} \end{bmatrix} = \begin{bmatrix} -K_p e_s - K_d \dot{e}_s - K_i \int e_s \mathrm{d}t \\ -K_p e_r - K_d \dot{e}_r - K_i \int e_r \mathrm{d}t \end{bmatrix} \tag{28}$$

Similarly, with the above PID controller, the second-order time-derivative of (26) shows:

$$\begin{cases} \ddot{e}_s + K_d \dot{e}_s + K_p e_s + K_i \int e_s \mathrm{d}t = d_s + \Delta_{c1} - \ddot{s}_{\text{ref}} \\ \ddot{e}_r + K_d \dot{e}_r + K_p e_r + K_i \int e_r \mathrm{d}t = d_r + \Delta_{c2} - \ddot{r}_{\text{ref}} \end{cases}$$

The controller gains are chosen such that the closed-loop system has slow dynamics since our blimp robot is supposed to move slowly. In this simulation, the gains are $K_p = 0.17$, $K_i = 0.01$, $K_d = 0.8$. From the above equation, we can see that for point stabilization task ($\ddot{s}_{\text{ref}} = \ddot{r}_{\text{ref}} = 0$), if the disturbances d_s and d_r are constant, the PID controller will make the steady-state error converging to zero. However, as it is mentioned before, the disturbance terms represent the errors between nominal model and real one, thus they certainly contain time-varying parts as set in our simulation. The point stabilization test result with PID controller is shown in Fig. 5. It is observed that the robot is oscillating around the desired position with a big error, thus the PID controller fails to compensate time-varying disturbance terms d_s and d_r.

Comparing the results obtained via the disturbance compensation based controller (Fig. 3) with that via PID controller (Fig. 5), we can say that the compensation of estimated disturbance terms in the designed controller helps to improve the robustness against external perturbations and system parameter identification inaccuracy.

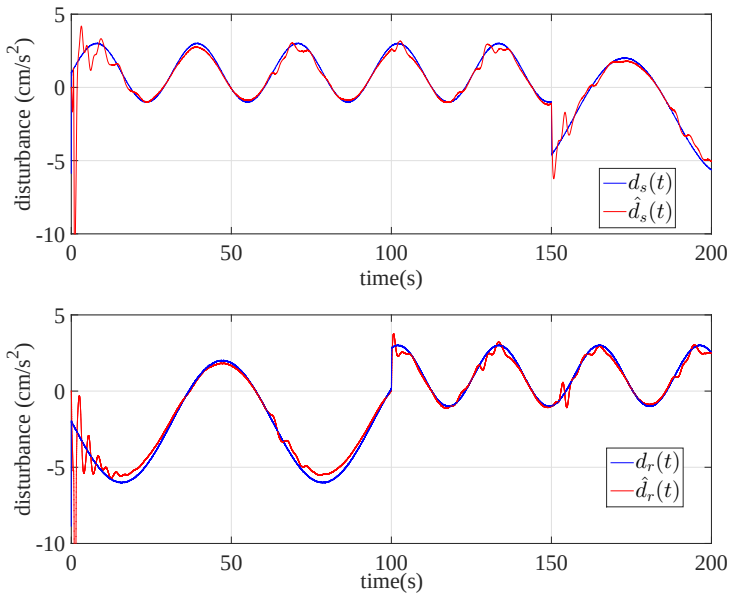

Figure 4: Point stabilization simulation Test-Disturbance d_s, d_r estimation result

5.3.2. Trajectory tracking

In this simulation test, the blimp is supposed to track a reference trajectory generated by the time-polynomial method. The initial state of blimp is set to be same as before:

$$s_i = r_i = 0, \quad \psi_i = -\pi/4$$

and at $t_f = 30$s, it is supposed to reach the goal position $s_f = -200, \quad r_f = 200$ then stabilized there. The trajectory $s_{\text{ref}}(t)$, $r_{\text{ref}}(t)$ are generated by third-order time-polynomials. The disturbance terms are set as:

$$d_x(t) = -1, \quad d_y(t) = 2, \quad d_\psi(t) = -0.2$$

Moreover a white-noise generated by "Gaussian Noise Generator" block of Simulink with variance set as 0.001 is added to d_x, d_y and d_ψ. The relation to obtain d_s and d_r is given by (20).

For the "real system" in simulation, the parameters are set to be the same as the nominal model, which means there is no parameter identification inaccuracy for this test.

The results of point Q tracking the reference trajectory are shown in Fig. 6 and 7.

It is shown that the blimp follows the desired trajectory with an acceptable accuracy, and even in the presence of disturbances, the controller robustly finishes its work.

The disturbance estimation result is given in Fig. 8.

It can be seen that when introducing the disturbance on the yaw angle d_ψ, the disturbances d_s and d_r depend also on the yaw angle, thus have more complex form, but the conceived disturbance estimator successfully evaluates the value, and helps to improve the controller robustness against perturbations.

From the simulation tests on point stabilization and trajectory tracking presented before, it can be concluded that the designed disturbance compensation based controller is able to achieve trajectory tracking of blimp in the horizontal plane and it is robust against small bounded disturbances. It is ready to be implemented on real blimp robot for validation.

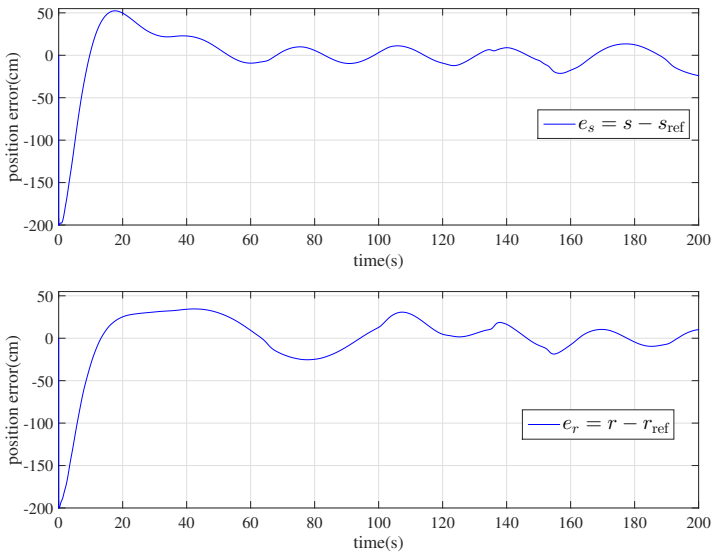

Figure 5: Point stabilization simulation Test with PID controller - Q position error result

6. Implementation and Results

Our developed blimp control board is shown in Fig. 9 and the complete robot is shown in Fig. 10.

The on-board micro-processor is STM32F103VET6 from STMicroelectronics [54], it is a powerful micro-processor which supports multiple functions and has good computation capability. However, due to the payload limit of the chosen balloon which is about 200 grams (see Fig. 10), there are only several dozens of grams left for low weight sensors to be carried on board, which raises the difficulties for autonomous localization in indoor environment for the robot. Therefore, the OptiTrack motion capturing system is installed in the testing room and integrated with robot control system to provide precise pose information of the robot.

6.1. Implementation with OptiTrack

The OptiTrack-enhanced blimp control system is shown in Fig. 11. It works as follows:

- First the OptiTrack system captures and tracks the blimp in indoor testing room, then solves blimp position and orientation and transmits the result via Ethernet to the host PC.

- Next, on host PC, a Simulink block diagram similar to the one designed for simulation test can be found, in which the blimp motion controller is implemented. However, the simulated "real system" is replaced by two interfaces: the one receives blimp pose information packet from OptiTrack, decodes the packet and extracts pose info; the other one packs motors commands into packet and sends it via wireless communication module XBee to blimp robot on-board micro-controller STM32.

- Then, on blimp robot control board, the XBee module receives packet from host PC, the micro-controller parses the packet and drives the motors by PWM waves with the help of motor driver board.

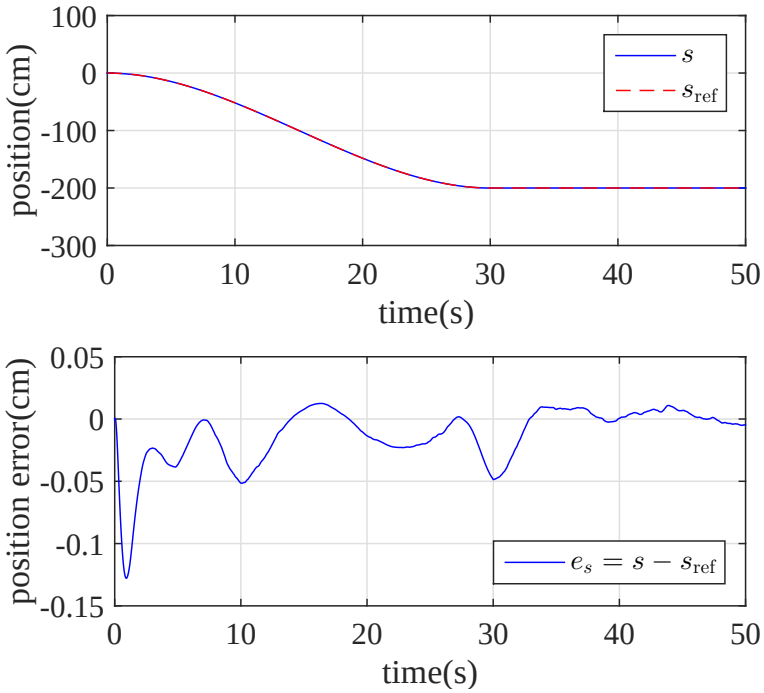

Figure 6: Trajectory tracking simulation test-Q position s and error e_s result

- Finally, the blimp is driven by the motors to reach desired goal, and its motion is always captured by OptiTrack system thus closing the system loop.

It is worth to mention that this scheme reuses the Simulink program designed during simulation for simplicity of testing and debugging. In real blimp indoor applications, we should implement all the conceived controllers into the robot on-board micro-controller, and the host PC only sends the mission commands to blimp. As presented before, the designed motion controllers are not complex; they both have only several hundreds lines of code, and do not require high computational capacity to solve the command signal. Moreover, the size of program is less than the Flash memory of chosen STM32 micro-controller. Therefore, the Simulink program can be easily transformed to be used on embedded microprocessor.

Although the OptiTrack system is easy to be used and provides high precision measurement of the robot pose, it has the problem of low flexibility, it is expensive and makes the robot not completely autonomous. In the future, we want to make the blimp robot more autonomous, localizing itself in unknown indoor environment only with on-board sensors. But at this moment, the OptiTrack-enhanced blimp control system is implemented for the validation of the designed control laws.

6.2. Results

As mentioned in Remark 2, the motion of blimp robot is decoupled into two independent parts, and controllers are conceived separately for the two motions. The validation of altitude motion controller was presented in [1], and in this paper the controller for planar movement is designed. Next, the two controllers can be combined together to achieve the full motion control of the blimp robot. Two types of tests are performed on the developed experiment platform to validate the controller and test its performance and robustness against perturbations.

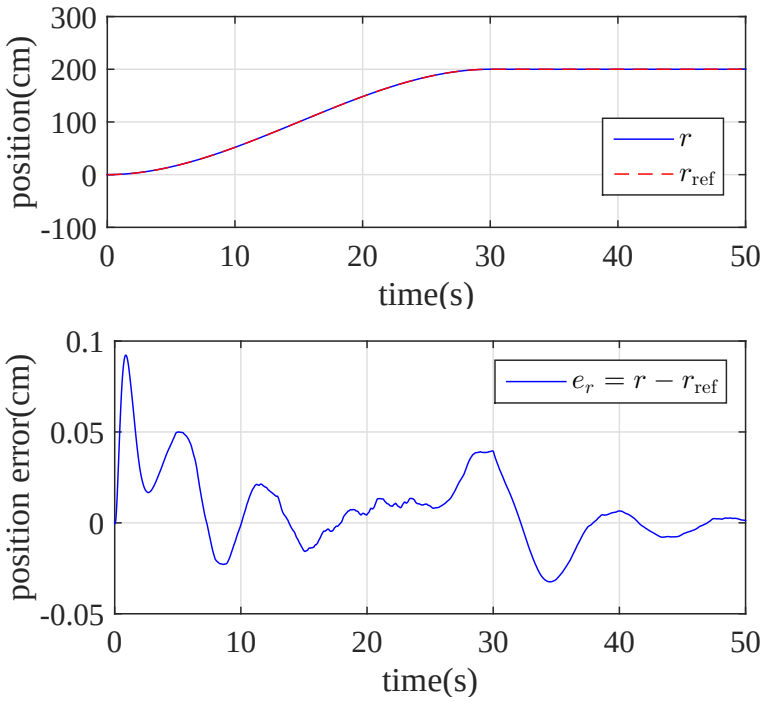

Figure 7: Trajectory tracking simulation test-Q position r and error e_s result

6.2.1. Point stabilization

The first one is to stabilize the blimp at a point in the space, which can be considered as a special case of trajectory tracking with constant s_{ref}, r_{ref} and z_{ref}. The reference point is set as (units: cm):

$$\begin{cases} s_{\text{ref}} = 60 \\ r_{\text{ref}} = -50 \\ z_{\text{ref}} = 300 \end{cases}$$

As it is discussed before, we achieve a practical control of the point Q instead of the exact control of blimp body-fixed frame center O_b. Therefore when the blimp is regulated to the goal position $(s_{\text{ref}}, r_{\text{ref}}, z_{\text{ref}})$, for the point O_b, its altitude is exactly regulated but on the horizontal plane its position lies practically on a circle centered at $(s_{\text{ref}}, r_{\text{ref}})$ and with a radius q (5cm in our setting). The result of blimp robot point stabilization is shown in Fig. 12.

It can be seen that the blimp reaches the goal point with position error less than 10cm within 20 seconds, which is acceptable considering the slow dynamics of the robot. Moreover, when we manually push the blimp horizontally at moment 50s and 110s, it returns to the set point. In addition, the system buoyancy force is bigger than gravity, but the altitude is still stabilized face to this constant disturbance.

The disturbance estimation results are shown in Fig. 13.

It is clear that the estimation $\hat{d}_s$ and $\hat{d}_r$ respond quickly to the external disturbances at moment 50s and 110s, thus help the controller to regulate the blimp back to its desired position, and the $\hat{d}_z$ also correctly estimates the constant external disturbances. It is worth to notice that although theoretically the altitude motion and planar movement of the blimp robot are decoupled, in reality they are not. The influences of one motion to another is also

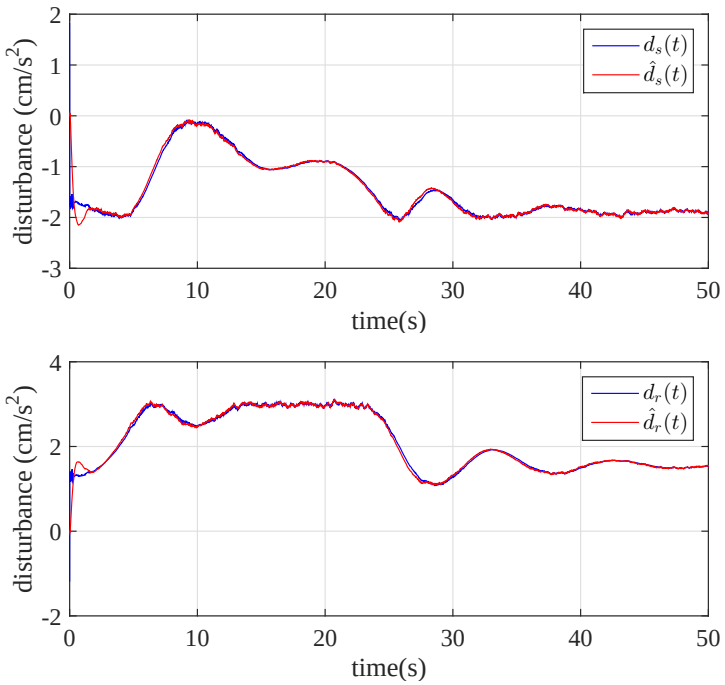

Figure 8: Trajectory tracking simulation test-disturbance estimation result

estimated by the designed algorithm, which can be observed at the first 30 seconds during the reaching phase of the blimp to its goal position.

The point stabilization test demonstrates the efficiency and robustness of the designed blimp robot motion controllers. It can be seen as a simulation scenario of the blimp monitoring a target room at a fixed location.

A video of this experiment can be found here: `https://drive.google.com/drive/folders/1Z0f-ZEppNcTuCIsqHWaAdpYQib6jpjF1?usp=sharing`.

6.2.2. Path following

In the second test, we want to make the blimp robot follow a designed path by choosing several intermediate points on the desired trajectory, when the blimp approaches closely enough the current way point, the goal reference point is set to the next intermediate point, and the controller continuously makes the blimp reaching the goal point. If a large number of intermediate points is chosen with temporal requirements, the process becomes similar to the trajectory tracking. But in our test, since the blimp is supposed to move slowly in the workspace, and the spatial errors are more critical than the temporal ones, thus only few key intermediate points are chosen for the blimp to move in an office room.

The intermediate points are chosen as (units: cm):

$$\begin{cases} W_1(s,r,z) = (30,0,180) \\ W_2(s,r,z) = (30,0,310) \\ W_3(s,r,z) = (130,0,310) \\ W_4(s,r,z) = (130,0,200) \\ W_5(s,r,z) = (130,-50,160) \\ W_6(s,r,z) = (130,-100,120) \end{cases}$$

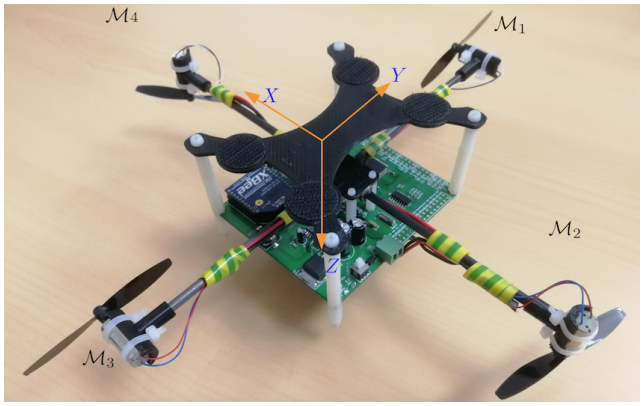

Figure 9: The developed blimp robot control board

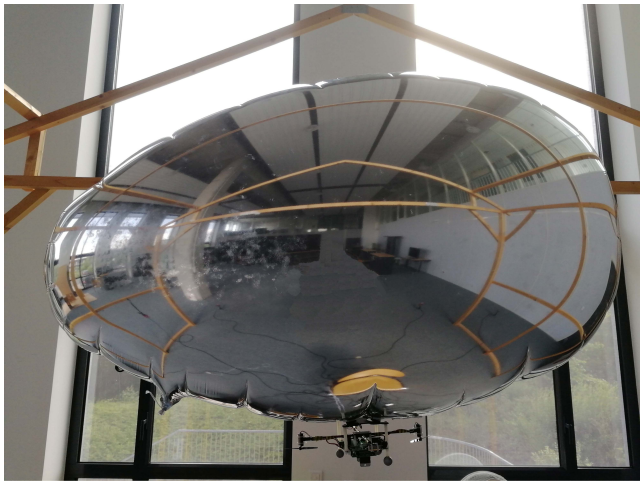

Figure 10: The blimp robot studied in this work

The path following result is shown in Fig. 14. Only the part from 20s to 77s is shown in the figure. For practical reasons, the threshold to judge whether the robot has reached a way point or not is set as 10cm, it is enough for our application scenario.

A video of this experiment can be found here: `https://drive.google.com/drive/folders/1rzP5nhhFL9jfXn_ipbE29lJ8-QfrcaPW?usp=sharing`.

This test is a simulation scenario of the blimp following a predefined path and monitoring a room. In practice, the blimp is set to pass through narrow space such as windows. In order to achieve that operation, for the blimp horizontal plane movement, as we have explained in Section 4, two controllers U_{xy} and U_{ψ} are used. The two controllers operate alternatively, which means when the blimp is far from the current reference point, the position regulation controller U_{xy} works; when it approaches the reference point with an acceptable tolerance, the yaw angle controller U_{ψ} is switched on to stabilize the yaw angle at desired direction. Then if both the position and heading angle are well stabilized, the blimp can start to move to its next goal, otherwise, the two controllers works alternatively to achieve the mission.

The video shows that the blimp moves smoothly and follows the designed way points with satisfying accuracy. Therefore, in real application scenarios, it should be able to finish the missions successfully.

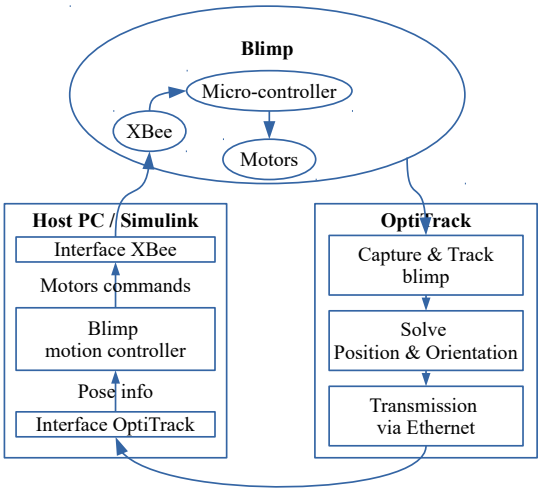

Figure 11: Scheme of Optitrack-enhanced blimp control system

7. Conclusion

In this paper, we presented the study of indoor small blimp robot robust controller design, including simplification and decoupling of the dynamic model, parameter identification of the simplified model, transformation of the planar model to a simpler form, disturbance compensation based controller design and its validation via simulation and real tests on the developed blimp robot. Experiments show the effectiveness and robustness of the designed controller for the task of point stabilization and path following in the presence of disturbances. Real test results are satisfying and they are similar to simulations.

Despite the encouraging results obtained, there are still many interesting and challenging topics which can be studied in the future. For instance, if we have access to more powerful and accurate testing devices, the model parameters can be re-identified with higher accuracy, and the added-mass terms along $\boldsymbol{X_b}$- and $\boldsymbol{Y_b}$-axis direction can be studied and considered when designing the controllers. In addition, we can study the use of flatness theory to design better trajectory for the blimp to track, and realize exact position and pose control of the robot instead of the practical one. Furthermore, the use of camera motion capturing system for the localization of the robot is not flexible and expensive, which limits the application of the blimp robot in more general indoor environments, in the future, we can try to integrate camera sensor on the robot to achieve fully autonomous navigation. It is worth to mention that the blimp robot is a perfect platform for the use of camera since it has the ability for VTOL, stationary and low-speed flight. Moreover, in the videos it is shown that the pitch and roll angles have slight oscillation when the robot is in motion, thus in the future, we should relax the assumptions, and design robust nonlinear controllers for the full motion control.

Acknowledge

This work is supported in part by ANR Finite4SoS (ANR 15 CE23 0007), by project Inventor (I-SITE ULNE, le programme d'Investissements d'Avenir, Mtropole Europenne de Lille), and by project VALID (CPER DATA), by the Government of Russian Federation [grant

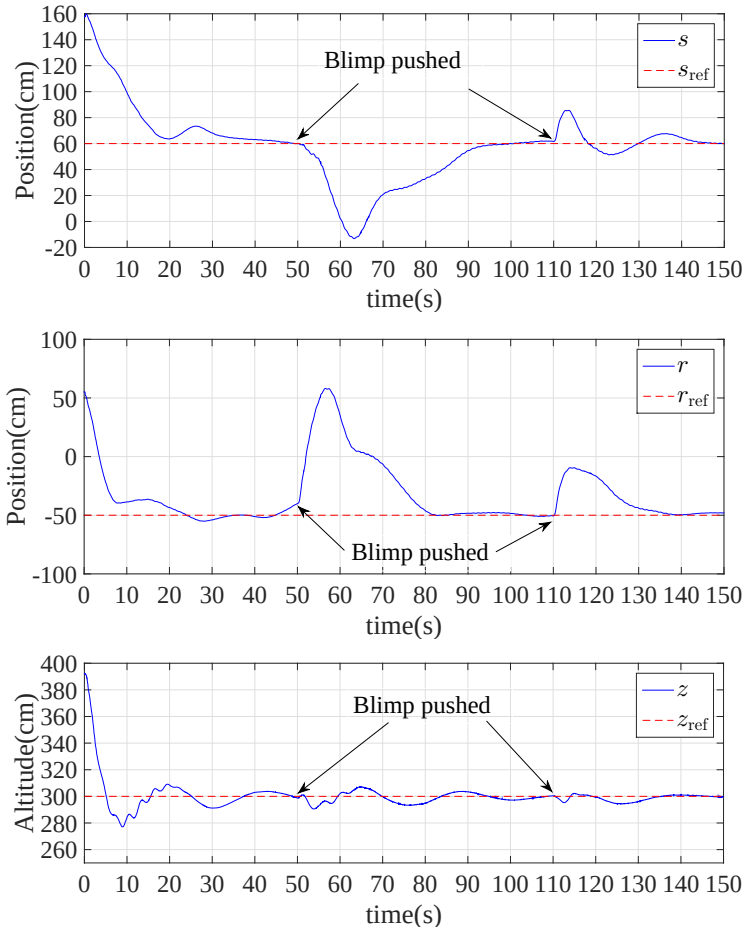

Figure 12: Blimp robot point stabilization test - position of control point $Q(s, r, z)$

number 074-U01] and the Ministry of Education and Science of Russian Federation [Project 14.Z50.31.0031].

Appendix: Explicit form of 6-DOF dynamic model

$$
\begin{aligned}
m'_x\dot{v}^b_x + m\dot{\omega}^b_y z_G - \omega^b_z(m'_y v^b_y - m\omega^b_x z_G) + m'_z\omega^b_y v^b_z - v^b_x(D_{v_x} + D_{v^2_x}|v^b_x|) + (f_G - f_B)\sin\theta = f_{px}\\
m'_y\dot{v}^b_y - m\dot{\omega}^b_x z_G + \omega^b_z(m'_x v^b_x + m\omega^b_y z_G) - m'_z\omega^b_x v^b_z - v^b_y(D_{v_y} + D_{v^2_y}|v^b_y|) - (f_G - f_B)\cos\theta\sin\phi = f_{py}\\
m'_z\dot{v}^b_z + \omega^b_x(m'_y v^b_y - m\omega^b_x z_G) - \omega^b_y(m'_x v^b_x + m\omega^b_y z_G) - v^b_z(D_{v_z} + D_{v^2_z}|v^b_z|) - (f_G - f_B)\cos\theta\cos\phi = f_{pz}\\
I'_x\dot{\omega}^b_x - m\dot{v}^b_y z_G + I'_z\omega^b_y\omega^b_z - v^b_z(m'_y v^b_y - m\omega^b_x z_G) - \omega^b_z(I'_y\omega^b_y + mv^b_x z_G) + m'_z v^b_y v^b_z - \omega^b_x(D_{\omega_x} + D_{\omega^2_x}|\omega^b_x|) + z_G f_G\cos\theta\sin\phi = \tau_{px}\\
I'_y\dot{\omega}^b_y + m\dot{v}^b_x z_G - I'_z\omega^b_x\omega^b_z + v^b_z(m'_x v^b_x + m\omega^b_y z_G) + \omega^b_z(I'_x\omega^b_x - mv^b_y z_G) - m'_z v^b_x v^b_z - \omega^b_y(D_{\omega_y} + D_{\omega^2_y}|\omega^b_y|) + z_G f_G\sin\theta = \tau_{py}\\
I'_z\dot{\omega}^b_z + v^b_x(m'_y v^b_y - m\omega^b_x z_G) - v^b_y(m'_x v^b_x + m\omega^b_y z_G) + \omega^b_x(I'_y\omega^b_y + mv^b_x z_G) - \omega^b_y(I'_x\omega^b_x - mv^b_y z_G) - \omega^b_z(D_{\omega_z} + D_{\omega^2_z}|\omega^b_z|) = \tau_{pz}
\end{aligned}
\tag{29}
$$

[1] Y. Wang, G. Zheng, D. Efimov, W. Perruquetti, Differentiator application in altitude control for an indoor blimp robot, International Journal of Control 91 (9) (2018) 2121–2130. arXiv:https://doi.org/10.1080/00207179.2018.1441549, doi:10.1080/00207179.2018.1441549.

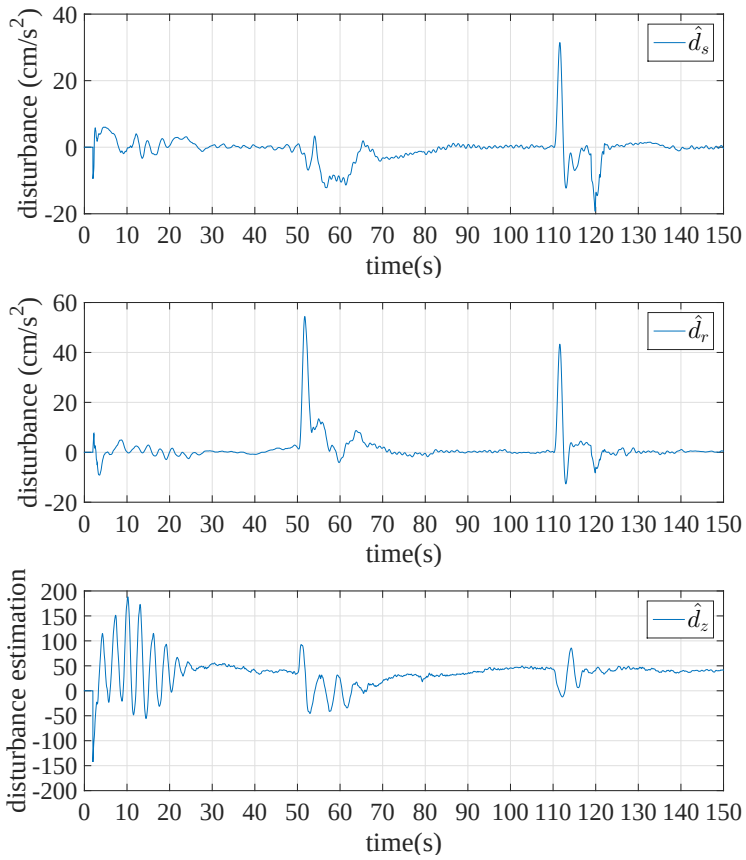

Figure 13: Blimp robot point stabilization test - disturbance estimation $\hat{d}_s$ (top), $\hat{d}_r$ (middle), $\hat{d}_z$ (bottom)

[2] X. Wang, Z. Zuo, C. Liu, Three dimensional path-following control of an under-actuated airship, in: 2016 35th Chinese Control Conference (CCC), 2016, pp. 4627–4632. doi:10.1109/ChiCC.2016.7554070.

[3] E. Kahale, P. C. Garcia, Y. Bestaoui, Autonomous path tracking of a kinematic airship in presence of unknown gust, Journal of Intelligent & Robotic Systems 69 (1-4) (2013) 431–446.

[4] L. Chen, J. F. Whidborne, Q. Dong, D. P. Duan, Application of lyapunov matrix inequality based unsymmetrical saturated control to a multi-vectored propeller airship, Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 232 (5) (2018) 884–901.

[5] L. Chen, D. Duan, Y. Wang, Lmi-based saturated control of a multi-vectored propeller airship, in: 2017 36th Chinese Control Conference (CCC), IEEE, 2017, pp. 6323–6329.

[6] Z. Zheng, Z. Guan, Y. Ma, B. Zhu, Constrained path-following control for an airship with uncertainties, Engineering Applications of Artificial Intelligence 85 (2019) 295–306.

[7] M. Zhu, Y. Song, Z. Zheng, W. Lou, Spatial adaptive integral los path following control for an under-actuated stratospheric airship, in: 2017 36th Chinese Control Conference (CCC), 2017, pp. 4912–4917.

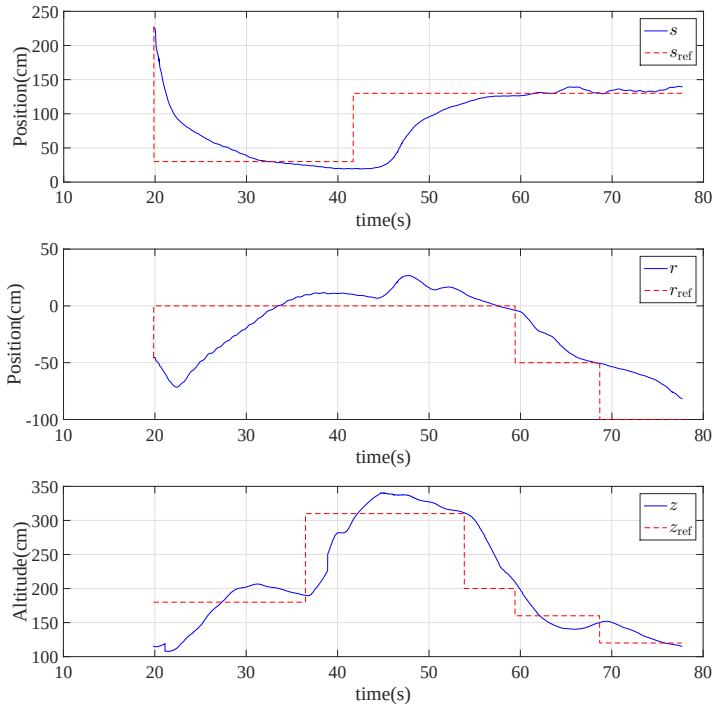

Figure 14: Blimp robot path following test - position of control point $Q(s, r, z)$

[8] Z. Zheng, Y. Zou, Adaptive integral los path following for an unmanned airship with uncertainties based on robust rbfnn backstepping, ISA transactions 65 (2016) 210–219.

[9] R. C. Do Valle, L. L. Menegaldo, A. M. Simões, Smoothly gain-scheduled control of a tri-turbofan airship, Journal of Guidance, Control, and Dynamics 38 (1) (2014) 53–61.

[10] A. Moutinho, J. R. Azinheira, E. C. de Paiva, S. S. Bueno, Airship robust path-tracking: A tutorial on airship modelling and gain-scheduling control design, Control Engineering Practice 50 (2016) 22–36.

[11] Y. Yang, Y. Yan, Trajectory tracking for robotic airships using sliding mode control based on neural network approximation and fuzzy gain scheduling, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 230 (2) (2016) 184–196.

[12] Y. Wang, P. Zhou, J.-A. Chen, D. Duan, Finite time attitude tracking control of an autonomous airship, Transactions of the Institute of Measurement and Control 40 (1) (2018) 155–162.

[13] M. Zhu, S. Yu, Z. Zheng, Model predictive control for path following of stratospheric airship with magnitude and rate constrains of rudder, in: The 27th Chinese Control and Decision Conference (2015 CCDC), IEEE, 2015, pp. 3444–3449.

[14] S. Yu, G. Xu, K. Zhong, S. Ye, W. Zhu, Direct-adaptive fuzzy predictive control for path following of stratospheric airship, in: 2017 29th Chinese Control And Decision Conference (CCDC), IEEE, 2017, pp. 5658–5664.

[15] E. Zhu, J. Pang, N. Sun, H. Gao, Q. Sun, Z. Chen, Airship horizontal trajectory tracking control based on active disturbance rejection control (adrc), Nonlinear Dynamics 75 (4) (2014) 725–734.

[16] R. Sanz, P. Garcia, Q.-C. Zhong, P. Albertos, Robust control of quadrotors based on an uncertainty and disturbance estimator, Journal of Dynamic Systems, Measurement, and Control 138 (7) (2016) 071006.

[17] G. Wyeth, I. Barron, An autonomous blimp, in: Field and service robotics, Springer, 1998, pp. 464–470.

[18] H. Zhang, J. P. Ostrowski, Visual servoing with dynamics: Control of an unmanned blimp, in: Robotics and Automation (ICRA), 1999 IEEE International Conference on, Vol. 1, IEEE, 1999, pp. 618–623.

[19] S. van der Zwaan, A. Bernardino, J. Santos-Victor, Vision based station keeping and docking for an aerial blimp, in: Intelligent Robots and Systems (IROS), 2000 IEEE/RSJ International Conference on, Vol. 1, IEEE, 2000, pp. 614–619.

[20] T. Fukao, K. Fujitani, T. Kanade, An autonomous blimp for a surveillance system, in: Intelligent Robots and Systems (IROS), 2003 IEEE/RSJ International Conference on, Vol. 2, IEEE, 2003, pp. 1820–1825.

[21] T. Fukao, K. Fujitani, T. Kanade, Image-based tracking control of a blimp, in: Decision and Control (CDC), 2003. Proceedings. 42nd IEEE Conference on, Vol. 5, IEEE, 2003, pp. 5414–5419.

[22] T. Fukao, T. Kanzawa, K. Osuka, Tracking control of an aerial blimp robot based on image information, in: Control Applications, 2007. CCA 2007. IEEE International Conference on, IEEE, 2007, pp. 874–879.

[23] T. Fukao, A. Yuzuriha, T. Suzuki, T. Kanzawa, T. Oshibuchi, K. Osuka, T. Kohno, M. Okuyama, Y. Tomoi, M. Nakadate, Inverse optimal velocity field control of an outdoor blimp robot–blimp surveillance systems for rescue–, IFAC Proceedings Volumes 41 (2) (2008) 4374–4379.

[24] H. Saiki, T. Fukao, T. Urakubo, T. Kohno, Hovering control of outdoor blimp robots based on path following, in: Control Applications (CCA), 2010 IEEE International Conference on, IEEE, 2010, pp. 2124–2129.

[25] W. E. Green, K. W. Sevcik, P. Y. Oh, A competition to identify key challenges for unmanned aerial robots in near-earth environments, in: Advanced Robotics, 2005. ICAR'05. Proceedings., 12th International Conference on, IEEE, 2005, pp. 309–315.

[26] S. B. i Badia, P. Pyk, P. F. Verschure, A biologically based flight control system for a blimp-based uav, in: Robotics and Automation (ICRA), 2005 IEEE International Conference on, IEEE, 2005, pp. 3053–3059.

[27] J.-C. Zufferey, A. Guanella, A. Beyeler, D. Floreano, Flying over the reality gap: From simulated to real indoor airships, Autonomous Robots 21 (3) (2006) 243–254.

[28] H. Fukushima, R. Saito, F. Matsuno, Y. Hada, K. Kawabata, H. Asama, Model predictive control of an autonomous blimp with input and output constraints, in: Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE, IEEE, 2006, pp. 2184–2189.

[29] H. Fukushima, K. Kon, Y. Hada, F. Matsuno, K. Kawabata, H. Asama, State-predictive control of an autonomous blimp in the presence of time delay and disturbance, in: Control Applications, 2007. CCA 2007. IEEE International Conference on, IEEE, 2007, pp. 188–193.

[30] J. Ko, D. J. Klein, D. Fox, D. Haehnel, Gaussian processes and reinforcement learning for identification and control of an autonomous blimp, in: Robotics and Automation (ICRA), 2007 IEEE International Conference on, 2007, pp. 742–747. doi:10.1109/ROBOT.2007.363075.

[31] A. Rottmann, C. Plagemann, P. Hilgers, W. Burgard, Autonomous blimp control using model-free reinforcement learning in a continuous state and action space, in: Intelligent Robots and Systems (IROS), 2007 IEEE/RSJ International Conference on, IEEE, 2007, pp. 1895–1900.

[32] A. Rottmann, T. Zitterell, W. Burgard, L. Reindl, C. Scholl, Towards an experimental autonomous blimp platform, in: Proc. of the Europ. Conf. on Mobile Robots (ECMR), 2007.

[33] J. Müller, A. Rottmann, L. M. Reindl, W. Burgard, A probabilistic sonar sensor model for robust localization of a small-size blimp in indoor environments using a particle filter, in: Robotics and Automation (ICRA), 2009 IEEE International Conference on, IEEE, 2009, pp. 3589–3594.

[34] J. Müller, N. Kohler, W. Burgard, Autonomous miniature blimp navigation with online motion planning and re-planning, in: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, IEEE, 2011, pp. 4941–4946.

[35] M. Burri, L. Gasser, M. Kach, M. Krebs, S. Laube, A. Ledergerber, D. Meier, R. Michaud, L. Mosimann, L. Muri, et al., Design and control of a spherical omnidirectional blimp, in: Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on, IEEE, 2013, pp. 1873–1879.

[36] N. Yao, E. Anaya, Q. Tao, S. Cho, H. Zheng, F. Zhang, Monocular vision-based human following on miniature robotic blimp, in: Proceedings - IEEE International Conference on Robotics and Automation, IEEE, 2017, pp. 3244–3249.

[37] D. St-Onge, P.-Y. Brèches, I. Sharf, N. Reeves, I. Rekleitis, P. Abouzakhm, Y. Girdhar, A. Harmat, G. Dudek, P. Giguère, Control, localization and human interaction with an

autonomous lighter-than-air performer, Robotics and Autonomous Systems 88 (2017) 165–186.

[38] T. I. Fossen, et al., Guidance and control of ocean vehicles, Vol. 199, Wiley New York, 1994.

[39] S. Gomes, An investigation into the flight dynamics of airships with application to the yez-2a, Ph.D. thesis, Cranfield Institute of Technology (10 1990).

[40] S. I. Sagatun, T. I. Fossen, Lagrangian formulation of underwater vehicles' dynamics, in: Systems, Man, and Cybernetics, 1991.'Decision Aiding for Complex Systems, Conference Proceedings., 1991 IEEE International Conference on, IEEE, 1991, pp. 1029–1034.

[41] H. Curtiss, D. Hazen, W. Putman, Lta aerodynamic data revisited, Journal of Aircraft 13 (11) (1976) 835–844, cited By 6. doi:10.2514/3.58719.

[42] S. F. Hoerner, Fluid-dynamic Drag: practical information on aerodynamic drag and hydrodynamic resistance, Hoerner Fluid Dynamics, 1958.

[43] B. D'Andra-Novel, S. Thorel, Control of non holonomic or under-actuated mechanical systems: The examples of the unicycle robot and the slider, Esaim Control Optimisation & Calculus of Variations 22 (4) (2016).

[44] M. Reyhanoglu, Exponential stabilization of an underactuated autonomous surface vessel, Automatica 33 (12) (1997) 2249–2254, cited By 153. doi:10.1016/S0005-1098(97)00141-6.

[45] I. Fantoni, R. Lozano, F. Mazenc, K. Pettersen, Stabilization of a nonlinear underactuated hovercraft, in: Decision and Control (CDC), 1999. Proceedings of the 38th IEEE Conference on, Vol. 3, IEEE, 1999, pp. 2533–2538.

[46] NaturalPoint, Optitrack for robotics, https://optitrack.com/motion-capture-robotics/ (Oct. 2018).

[47] R. W. Brockett, Asymptotic stability and feedback stabilization, Differential geometric control theory 27 (1) (1983) 181–191.

[48] M. Guerra, D. Efimov, G. Zheng, W. Perruquetti, Avoiding local minima in the potential field method using input-to-state stability, Control Engineering Practice 55 (2016) 174–184.

[49] J. H. Lumkes Jr, Control strategies for dynamic systems: design and implementation, CRC Press, 2001.

[50] Y. Wang, G. Zheng, D. Efimov, W. Perruquetti, Improved altitude control method with disturbance compensation for an indoor blimp robot, in: 2017 IEEE 56th Annual Conference on Decision and Control (CDC), Vol. 2018-January, 2018. doi:10.1109/CDC.2017.8264233.

[51] A. Levant, Higher-order sliding modes, differentiation and output-feedback control, International Journal of Control 76 (9-10) (2003) 924–941.

[52] W. Perruquetti, T. Floquet, E. Moulay, Finite-time observers: application to secure communication, IEEE Transactions on Automatic Control 53 (1) (2008) 356–360.

[53] C. Aguilar-Ibez, H. Sira-Ramrez, M. Surez-Castan, E. Martnez-Navarro, M. Moreno-Armendariz, The trajectory tracking problem for an unmanned four-rotor system: Flatness-based approach, International Journal of Control 85 (1) (2012) 69–77. doi:10.1080/00207179.2011.638328.

[54] STMicroelectronics, Stm32f103ve product, https://www.st.com/en/microcontrollers/stm32f103ve.html (Dec. 2018).